Fast Video Object Segmentation via Dynamic Targeting Network

Lu Zhang[1], Zhe Lin[2], Jianming Zhang[2], Huchuan Lu[1]*, You He[3]
[1]Dalian University of Technology, China
[2]Adobe Research, USA
[3]Naval Aviation University, China
luzhang_dut@mail.dlut.edu.cn, {zlin, jianmzha}@adobe.com, lhchuan@dlut.edu.cn, heyou_f@126.com

Abstract

We propose a new model for fast and accurate video object segmentation. It consists of two convolutional neural networks, a Dynamic Targeting Network (DTN) and a Mask Refinement Network (MRN). DTN locates the object by dynamically focusing on regions of interest surrounding the target object. The target region is predicted by DTN via two sub-streams, Box Propagation (BP) and Box Re-identification (BR). The BP stream is faster but less effective at objects with large deformation or occlusion. The BR stream performs better in difficult scenarios at a higher computation cost. We propose a Decision Module (DM) to adaptively determine which sub-stream to use for each frame. Finally, MRN is exploited to predict segmentation within the target region. Experimental results on two public datasets demonstrate that the proposed model significantly outperforms existing methods without online training in both accuracy and efficiency, and is comparable to online training-based methods in accuracy with an order of magnitude faster speed.

1. Introduction

Video object segmentation (VOS) aims to segment target objects across video frames. It is a challenging task due to the motion, occlusion and deformation of objects. Given the first frame with mask annotations, our task is to track the specific objects in the following video frames, which is known as semi-supervised VOS.

Recently, semi-supervised VOS has achieved impressive progress thanks to the advances of Convolutional Neural Networks (CNNs). Precise object segmentation is able to facilitate the performance of various applications, such as video object tracking, surveillance and interactive video editing. Except for the demand of high accuracy, the processing efficiency of the algorithms is also required in time-critical applications. To achieve a well-performed model, existing CNN-based methods usually conduct online training, in which the pretrained models are finetuned on the first frame of the given test video. The online training is effective at improving the model's generalization to appearance variations of target objects. However, it incurs a significant computation overhead and thus limits the applications of the existing VOS models in time-critical scenarios.

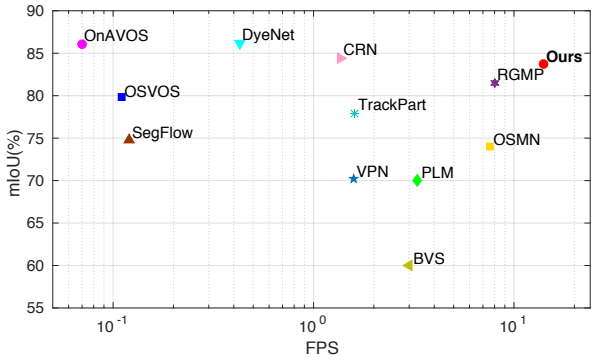


Figure 1: Comparison results with state-of-the-art methods on the DAVIS 2016 dataset in terms of $\mathcal{J}$ Mean (mIoU) and run time (fps).

Some recent works [23, 2, 3, 5] combine the merits of object re-identification and instance segmentation for video object segmentation. Typically, the target RoI in the current frame is re-identified by matching the box candidates with the annotated object in the first frame. Then the mask is further segmented out within the selected RoI. Such object re-identification mechanism has shown promising effectiveness in dealing with object occlusion or drifting. However, the matching process among hundreds of candidates often brings large computation cost, thus limiting its use in real-time applications.

In this work, we propose a new model for fast and accurate video object segmentation. Our model consists of two sub-networks with a shared backbone, which are Dynamic Targeting Network (DTN) and Mask Refinement Network (MRN). Specifically, we first exploit the DTN to automatically zoom in to the potential region of the target object.

*Corresponding author

Then the MRN is used to predict accurate segmentation mask within the target region in a coarse to fine manner.

The dynamic targeting network contains two sub-streams, Box Propagation (BP) stream and Box Re-identification (BR) stream, for producing target RoI from different aspects. In the BP stream, we exploit the temporal continuity via optical flow [16] to efficiently propagate the box coordinates between adjacent frames. Despite the fast processing time, BP has limited ability in handling objects with occlusion or large deformation. To this end, we propose the BR stream in which the target box is re-identified from a set of candidates generated by Conditional Region Proposal Network (CRPN). Compared with BP, BR is slower but more robust for complicated scenes. To achieve an equilibrium between segmentation accuracy and computation efficiency, we further propose a switchable architecture to automatically pick a sub-stream for each incoming video frame. Specifically, we first use a Decision Module (DM) to produce a confidence score for each frame, which reflects whether the BP stream could generate the correct box. The frames whose confident scores are higher than a fixed threshold can go through the BP stream to generate target box and vice versa. The DTN can flexibly achieve various trade-offs between accuracy and efficiency by adjusting the value of the confidence score.

Given the potential target region, we then leverage mask refinement network to produce the corresponding segmentation mask. We first use RoI Align [11] to extract multi-level features for the target RoI. The object mask in the last frame is warped by optical flow to serve as a prior guidance. We then refine it in a coarse to fine way to generate the current target mask. To verify the effectiveness of the proposed model, we conduct experiments including overall comparisons and ablation studies on the DAVIS dataset [29, 30]. The results show that the proposed model significantly outperforms existing methods without online learning, and is comparable with the online training-based methods.

Our contributions can be summarized as follows:

- We propose a new method by seamlessly integrating target RoI generation and mask prediction for video object segmentation.
- We propose a novel decision module to dynamically assign frames to two sub-stream networks (box re-identification and box propagation) to allow balancing/prioritizing between accuracy and efficiency.
- We perform experiments to show that our model significantly outperforms existing methods without online training, and is comparable to the online training-based methods in accuracy at a much faster speed.

2. Related Work

Unsupervised video object segmentation. Unsupervised video object segmentation (Un-VOS) models focus on segmenting the foreground objects within the whole video without any manual annotations. Previous methods usually exploit visual saliency [33, 15] or motion cues [19, 23, 18, 22] to obtain prior information of the prominent objects. Recently, some CNN-based methods [23, 4] have shown impressive performance by using rich features and large training datasets. However, the Un-VOS methods could not be applied to segment a specific object due to the ambiguous definition of foreground object via motion.

Semi-supervised video object segmentation. Semi-supervised video object segmentation (Semi-VOS) aims to segment the objects specified by users in the first frame. Inspired by the successful applications of CNNs on image segmentation [8, 7, 6, 40, 39], many deep learning based methods have been proposed for semi-VOS and shown impressive performance. These approaches can be divided into two categories, propagation-based and detection-based.

The propagation-based methods [17, 14, 34, 28, 1, 26] exploit deep network to implicitly model the motion information and propagate the segmentation frame by frame from the first annotation. For example, [17] proposes a unified framework of temporal bilateral network and spatial refinement network for adaptively propagating structure information through the entire video. In [14], Hu *et al.* present a recurrent network to propagate the mask and bounding box from previous outputs simultaneously. Perazzi *et al.* [28] build a MaskTrack model, where the results are calculated based on both the current frame and the previous mask. The continuous offline training and online finetuning boost the performance of the network. A recent work [34] puts forward a siamese network to utilize make guidance from both previous and the first frames for mask propagation.

Another category in Semi-VOS is the detection-based methods, which exploit referring frame as a target to detect the object mask in each frame. [2] proposes a one-shot online learning strategy, in which the pre-trained network is finetuned with the first annotated frame for each test video. In [31], Yoon *et al.* use multi-level CNN features for calculating the pixel-level similarity between testing and reference frames. Some methods propose to utilize both initial and previous frame to provide more reliable references. Yang *et al.* [35] put forward to use referring annotation and spatial prior from the last frame to automatically modulate the network. In [3], Cheng *et al.* leverage tracker and RoI segmentation network to localize and segment the partial regions of the object, which are further fused to generate the final segmentation. Li *et al.* [23] propose a model to jointly re-identify the object and temporally propagate the mask along the entire video. In this paper, we propose a

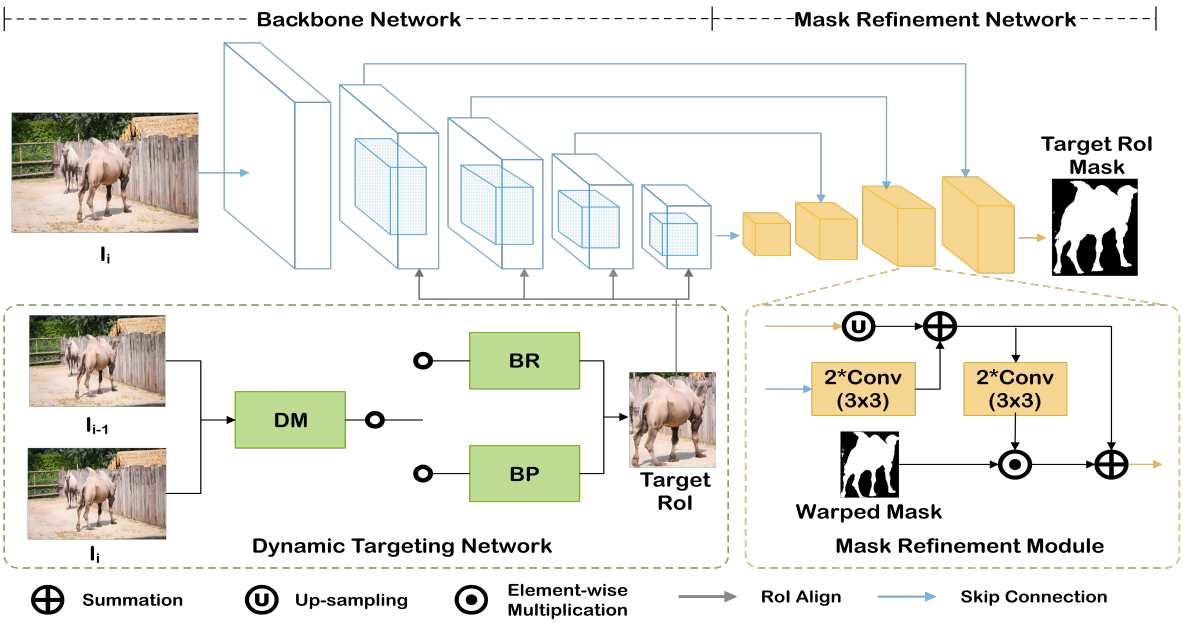


Figure 2: An overview of our proposed method. Our model contains two sub-networks, Dynamic Targeting Network (DTN) and Mask Refinement Network (MRN). For the *i-th* frame, we first exploit Resnet50 [12] to extract multi-level features. Then the DTN automatically generates a potential foreground RoI, which is further processed by MRN to obtain its object mask. The DTN consists of a faster Box Propagation (BP) stream and a more accurate but slower Box Re-identification (BR) stream. We use a Decision Module (DM) to formulate a confidence score to assign frames into different sub-streams. The MRN takes the object mask of the last frame (warped by optical flow) as input and refines it using four stacked mask refinement modules.

new model for fast and accurate Semi-VOS that segments the object via dynamically focusing on regions of interest surrounding the target.

Dynamic network. The core idea of the dynamic network is to adaptively conduct different processing for different image regions or video frames. It can accelerate the processing speed while maintain good performance, and has been applied to many video-level tasks [24, 37, 36]. In [36], Zhu *et al.* exploit deep network to extract the features for the key frames and propagate them to nearby frames via a fast flow network. In [24], Li *et al.* propose an adaptive scheduler for key frame selection and use spatially variant convolution for feature propagation. Xu *et al.* [37] propose a decision network for assigning different regions to either a faster flow network or a slower segmentation network. In this paper, we propose a dynamic targeting network, in which different frames are assigned to different sub-streams for generating target object regions.

3. Method

3.1. Overall Architecture

An overview of our model architecture is illustrated in Fig. 2. It consists of two sub-networks, Dynamic Targeting Network (DTN) and Mask Refinement Network (MRN). We use a shared backbone network to produce general features for both DTN and MRN. Specifically, we choose Resnet50 [12] as the shared feature extractor due to its well-balanced capacity and efficiency. Given an input frame sequence $I_i \in \{I_1, ..., I_N\}$, we employ the last output from Res2_x to Res5_x to construct the multi-level features, which are represented as $\mathbf{F}_i = \{\mathbf{f}_i^j\}_{j=2}^5$. We then use the proposed DTN to automatically generate a zoomed-in RoI which potentially contains the target object specified in the first frame. The DTN is designed as a switchable two-stream architecture with Box Propagation (BP) stream and Box Re-identification (BR) stream. The BP leverages temporal consistency between adjacent frames to conduct box propagation. It is faster but less accurate when handling objects with large deformation or occlusion. While the BR has better performance in such difficult cases, it is slower due to the burdensome box re-identification. To balance the accuracy and efficiency, we propose a Decision Module (DM) which learns to predict a confidence score to decide which stream each frame will pass through. Specifically, the frames whose confidence scores are higher than a predefined threshold will go through the faster BP stream and vice versa. Once we obtain the RoI of the target object in the current frame via DTN, we use MRN to refine the details of the RoI in a coarse to fine manner to obtain the final segmentation mask. In the following sections, we will introduce the details of DTN, MRN, and their training strategy, respectively.

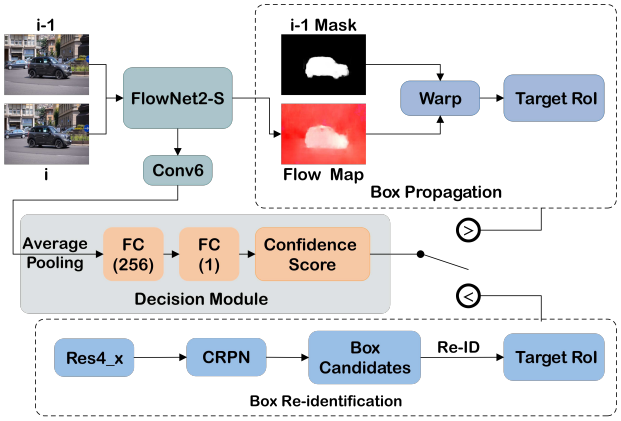


Figure 3: The framework of dynamic targeting network (DTN). The DTN has two sub-streams, box propagation (BP) and box re-identification (BR), for generating target RoI. The BP exploits optical flow for propagating box from the last frame, while the BR proposes a conditional RPN (CRPN) for box re-identification. The decision module is used to calculate confidence score for assigning frames to different streams. The frames with larger confidence scores than a threshold would be passed to BP, vice verse.

3.2. Dynamic Targeting Network

Some previous works utilize the advances in video tracking [3, 5] or object re-identification [23] to first attend to the target object. They have demonstrated that such methodology is beneficial for improving the accuracy in both object localization and segmentation. The above-mentioned two types of approaches have their own advantages. For example, the trackers [3, 5] have faster computation speed but might be unstable to localize objects with occlusion or fast movement. The detector [23] is more robust to deal with such difficult cases, but is slower due to the re-identification among hundreds of candidates. To balance the accuracy and efficiency, we propose a dynamic targeting network (DTN), in which frames are adaptively assigned to different sub-streams for generating target RoI. Our DTN has a decision module (DM) with two sub-streams, *i.e.,* box propagation (BP) stream and box re-identification (BR) stream. The BP is a fast and light network, which leverages optical flow [16] to propagate bounding box between adjacent frames. While the BR is a more computational-cost network, in which the target boxes are re-identified from the candidate set via feature matching. For the current input I_i, the goal of DTN is to produce K target RoIs $\{T_i^k\}_{k=1}^K$ that match with the K annotated objects in the first frame. The framework of DTN is shown in Fig. 3

Decision module. A recent work [37] proposes that dividing frame regions into various networks could achieve a good trade-off between speed and accuracy. A decision network [37] is put forward to determine that the simple regions in a frame are processed with a faster network, while the hard ones are fed to a more precise but slower network. Inspired by [37], we build a similar decision module (DM) to justify whether the faster box propagation stream could generate convincing target RoI for the current frame. Since the BP stream leverages optical flow to formulate box propagation, we construct the DM on the fast and well-performed FlowNet2-S [16]. Specifically, we first feed the current frame I_i along with the last frame I_{i-1} into the FlowNet2-S [16]. Our DM takes the features from the Conv6 of the FlowNet2-S as input. Then we exploit an average pooling layer and two fully connected layers to predict a confidence score C_i. The confidence score aims to indicate whether the flow-based BP stream is capable of producing proper target RoI for the current frame I_i.

For training DM, we define a ground truth confidence score as follows,

$$\widehat{C}_i = \frac{1}{K}\sum_{k=1}^{K}\mathcal{M}(T_i^k, \widehat{T}_i^k) \tag{1}$$

where $\widehat{C}_i$ is the ground truth confidence score. T_i^k is the predicted target RoI by BP stream and $\widehat{T}_i^k$ is the ground truth target RoI of frame I_i. $\mathcal{M}$ indicates the formulation of Intersection over Union (IoU) between T_i^k and $\widehat{T}_i^k$. The DM is trained with mean squared error (MSE) loss between predicted confidence score and the corresponding ground truth. During inference, DM compares the predicted confidence score with a pre-defined threshold θ_c. If it is higher than θ_c, the current frame would be fed to BP stream for fast box propagation from the last frame. Otherwise, it would be passed to BR stream. Fig. 4 shows the confidence scores of various frames, which verify that our DM can adaptively determine the proper sub-stream for each frame.

Box propagation stream. We propose to build a fast box propagation between adjacent frames. The optical flow [16] is exploited to capture the object motion between two frames. We begin with feeding the current frame I_i and the last frame I_{i-1} to Flownet2-S [16] to generate their optical flow map O_i. Note the flow net is shared in both decision module and box propagation stream. Then we take the predicted binary mask of the last frame $\{Y_{i-1}^k\}_{k=1}^K$ and warp it to $\{Y_{(i-1)\to i}^k\}_{k=1}^K$ according to flow map O_i with bilinear operation. Finally, we obtain the bounding boxes of the warped mask $\{Y_{(j-i)\to j}^k\}_{k=1}^K$ and take them as the target RoIs $\{T_i^k\}_{k=1}^K$ for the current frame I_i.

Box re-identification stream. The box propagation stream has a fast computation speed. However, the performance of optical flow might be influenced by object occlusion, fast movement or large deformation. This would drop the precision of box propagation stream for predicting target RoIs. To this end, we also propose a box re-identification stream for better handling such complicated scenarios. The re-identification process aims to find out the box candidates that are best-matched with the annotated objects in the first frame. A previous work [23] exploits Region Proposal Net-

Figure 4: Visual samples for dynamic targeting network. We show some input frames along with their optical flow maps. The tagged values indicate the frame indexes and the corresponding confidence scores predicted by decision module. The frames with smaller confidence scores mean that the box propagation is not convincing, and they would go through the box re-identification stream.

work (RPN) [11] to generate a set of box candidates. However, the matching process is performed among hundreds of proposals, which is very time-consuming. To solve this problem, we put forward a Conditional RPN (CRPN), in which the generation of anchors in the current frame are conditioned on the prediction of previous frame. An outline of our CRPN and the original RPN is shown in Fig. 5.

Given a feature map $\mathbf{f}_i^j$ with resolution $h_j \times w_j$, the original RPN [14] proposes anchors at each location with three scales and aspect ratios (See Fig. 5 (c)). Such large number of proposals ($h_i \times w_i \times 9$) would burden the computation cost of re-identification. While, in our CRPN, the area, scales and aspect ratios are calculated based on the output $\{Y_{i-1}^k\}_{k=1}^K$ of last frame. Specifically, we first obtain the center coordinates $\{l_{i-1}^k\}_{k=1}^K$, scales $\{s_{i-1}^k\}_{k=1}^K$ and aspect ratios $\{r_{i-1}^k\}_{k=1}^K$ from the last target RoIs $\{T_{i-1}^k\}_{k=1}^K$. Usually, the location and shape of objects would not change largely between two adjacent frames. In the current feature map $\mathbf{f}_i^j$, the anchors are predicted in a 3×3 grid centered at l_{i-1}^k (the gray grid in Fig 5 (b)). Meanwhile, the current scales s_i^k and aspect ratios r_i^k are defined as $s_i^k = \{(j) \times 0.5 s_{i-1}^k\}_{j=1}^3$ and $r_i^k = \{(j) \times 0.5 r_{i-1}^k\}_{j=1}^3$. As such, the number of anchors produced in our CRPN decreases to 9×9 for each target object. Our CRPN could largely fasten the box re-identification process and help to remove the noise boxes for better re-identification. The example of anchor generation in our CRPN is shown in Fig. 5 (b). With these anchors, we then predict the objectness scores and box regression for producing bounding boxes like the original RPN [11].

Our CRPN is built after the Res4_x. After obtaining the box candidates $\{B_i^{k,j}\}_{j=1}^{N^k}, k = 1, .., K$, we compare them with the target object by calculating a matching score,

$$S_j^k = \mathcal{D}(f(B_i^{k,j}), f(T_1^k)) + \mathcal{D}(f(B_i^{k,j}), f(T_{i-1}^k)) \quad (2)$$

S_j^k is the matching score for the j-th candidates $B_i^{k,j}$. $\mathcal{D}$ indicates L2 distance metric. $f(*)$ is the feature vector of bounding box, which is the average pooling of feature map generated by RoI Align [11] on Res4_x. Eq. 2 indicates that the matched candidate should have large feature similarity with the target object in both 1-th and $(i-1)$-th frames. We choose the candidates with the largest score as the target RoIs $\{T_i^k\}_{k=1}^K$ in the current frame.

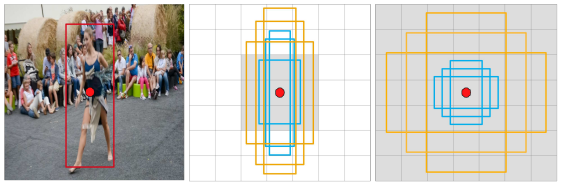


Figure 5: The implementation details of our CRPN. From left to right: (a) the (*i-1*)-th frame and target RoI $\{T_{i-1}^k\}_{k=1}^K$, (b)-(c) anchor generation of our CRPN and the original RPN. The gray grid denotes the location for anchor formulation. In our CRPN, anchor boxes are generated at the 3×3 area surrounding the center of last target RoI. Besides, the aspect ratios and scales are calculated based on the last target RoI. The number of proposals in our CRPN is $9 \times 9 \times K$ vs $h_j \times w_j \times 9$ in the original RPN ($K \times 9 << h_j \times w_j$).

3.3. Mask Refinement Network

With the target RoIs from DTN, the next step is to segment the corresponding object masks. We propose a Mask Refinement Network (MRN), which leverages multi-level features to enhance the details in a coarse to fine manner. Specifically, we use the features from Res2_x to Res5_x (denoted as $\{\mathbf{f}_i^j\}_{j=2}^5$) for mask generation. For the feature at j-th level, we extract the feature maps for target RoIs $\{T_i^k\}_{k=1}^K$ and resize them to $m_j \times m_j$ using RoI Align [11]. We set $m_j = \frac{M}{2^{(j-2)}}, j = 2, ..., 5$ to meet the feature resolution in different layers. We use the warped masks $\{Y_{(i-1)\to i}^k\}_{k=1}^K$ from the last output as prior maps and stack four mask refinement modules (MRM) to fuse them with the multi-level RoI features. The details of MRM is illustrated in Fig. 2. First, the RoI feature in j-th layer is combined with the output of the last MRM by

$$f_j^U(T_i^k) = Conv^2(f_j(T_i^k) + Up(f_{j+1}^M(T_i^k))) \quad (3)$$

where $f_j(T_i^k)$ is the RoI feature of T_i^k at level j. $f_{j+1}^M(T_i^k)$ indicates the output of the last MRM. $Up()$ is the upsampling operation with stride 2. $Conv^2()$ indicates two convolutional layers with 3×3 kernel size. Then we exploit the

Table 1: Overall comparison with state-of-the-arts on DAVIS-2016 validation dataset. We use "✓" to indicate whether the method is constructed with Online Finetuning (OF) or Post-processing (PP).

Method	OF	PP	$\mathcal{J}$ Mean	$\mathcal{F}$ Mean	Time
OnAVOS [32]	✓	✓	86.1	84.9	13s
OSVOS [2]	✓	✓	79.8	80.6	9s
DyeNet [23]	✓		**86.2**	-	2.3s
PLM [31]	✓	✓	70.0	62.0	0.3s
SegFlow [4]	✓		74.8	74.5	7.9s
MaskRNN [14]	✓		80.7	80.9	-
Lucid [20]	✓	✓	84.8	82.3	40s
MoNet [35]	✓	✓	84.9	84.8	14.4s
CRN [13]	✓		84.4	**85.7**	0.73s
BVS [27]			60.0	58.8	0.37s
VPN [17]			70.2	65.5	0.63s
RGMP [34]			81.5	82.0	0.13s
TrackPart [3]	✓		77.9	76.0	0.6s
OSMN [38]			74.0	72.9	0.14s
Ours			**83.7**	**83.5**	**0.07s**

warped object mask $Y^k_{(i-1)\to i}$ (refer to the box propagation stream in Sec. 3.2) as a prior to guide the mask segmentation for the current frame, which is conducted by

$$f^M_j(T^k_i) = Conv^2(f^U_j(T^k_i)) \odot Y^k_{(i-1)\to i} + f^U_j(T^k_i) \quad (4)$$

$f^M_j(T^k_i)$ is the output feature map of the j-th MRM and $\odot$ means element-wise multiplication. Note that the warped object mask $Y^k_{(i-1)\to i}$ should be resized to the resolution of the current feature $f^M_j(T^k_i)$. By stacking four MRMs, the features from deeper layers are gradually aggregated with the ones in shallower layers (*i.e.,* from Res5_x to Res2_x). To obtained the object masks $\{Y^k_i\}^K_{k=1}$ of the current target RoIs $\{T^k_i\}^K_{k=1}$, we feed the output of the last MRM into a 3×3 convolutional layer with sigmoid activation function.

3.4. Training and Inference

Implementation details. We exploit Resnet50 [12] as the backbone network and Flownet2-S [16] for formulating optical flow. For the decision module in the dynamic targeting network (see Sec. 3.2), we set the threshold θ_c as 0.83 to balance the accuracy and efficiency. The channel size of two fully connected layers in DM is set to 256 and 1, respectively. In the box re-identification stream, the features from Res4_x are used for box classification and regression of CRPN. For feature matching, we utilize the RoI Align [11] to produce 7×7 feature maps for each box candidates $B^{k,j}_i$. In the mask refinement network, the size of features by RoI Align is set to $m_j = \frac{112}{2^{(j-2)}}, j = 2, ..., 5$ for each feature level. The channel number of the convolutional layers in the mask refinement module is set to 256.

Training. The overall loss of our model is defined as follows:

$$L = L_{DM} + L_{CRPN} + L_{MRN} \quad (5)$$

where L_{DM} is the mean squared error loss between predicted confidence score C_i with ground truth confidence score $\widehat{C}_i$ defined in Eq. 1. L_{CRPN} is the bounding box classification and regression losses with the same definition as [11]. L_{MRN} indicates the cross entropy loss between predicted masks $\{Y^k_i\}^K_{k=1}$ and ground truth mask $\{\widehat{Y}^k_i\}^K_{k=1}$. Following the previous VOS methods [23, 34, 20], we pre-train our backbone network, the original RPN and mask refinement network on instance object segmentation task. Specifically, we exploit the instance masks from MSCOCO [25] and PASCAL VOC [9, 10]. We conduct data augmentation on both datasets with image flip, random rotation and crop. The input size is set to 512×512 in the pre-training stage. The three networks are trained using SGD with an initial learning rate 0.0025, batch size 2 and momentum 0.9. The learning rate is decrease by 0.1 in every 50k iterations.

After the pre-training stage on static images, we further finetune our model on DAVIS training set [29, 30]. In this stage, all the modules including backbone network, dynamic targeting network, and mask refinement network are jointly trained. We exploit the loss defined in Eq. 5 to train our model. We use SGD optimizer with a fixed learning rate 0.0001, batch size 1 and momentum 0.9. The data augmentation is also conducted in video dataset.

Inference. For each testing video sequences $\{I_i\}^N_{i=1}$, the ground truth masks of target objects are provided in the first frame. The subsequent frames with original size are fed into the model for producing object masks. Our model does not conduct the online training on the first frame of the test videos [2, 20].

Table 2: Overall comparison with state-of-the-arts on DAVIS-2017 validation dataset.

Method	OF	PP	$\mathcal{J}$ Mean	$\mathcal{F}$ Mean	$\mathcal{G}$ Mean
OnAVOS [32]	✓	✓	61.6	69.1	65.4
OSVOS [2]	✓	✓	56.6	63.9	60.3
MaskRNN [14]	✓		60.5	-	-
RGMP [34]			**64.8**	68.6	66.7
TrackPart [3]	✓		54.6	61.8	58.2
OSMN [38]			52.5	57.1	54.8
Ours			64.2	**70.6**	**67.4**

4. Experiment

4.1. Dataset and Metrics

Dataset. To validate the effectiveness of our proposed model, we conduct our experiments on DAVIS benchmarks [30, 29]. The DAVIS 2016 dataset [29] consists of 50 high-quality videos, in which total 3455 frames are annotated with densely pixel-wise object masks. The 50 video

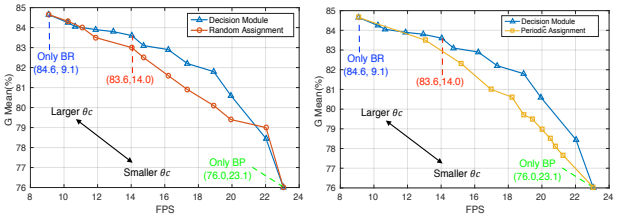


Figure 6: The trade-off between accuracy ($\mathcal{G}Mean$) and frame rate (fps) under various thresholds (θ_c) in decision module. The curves are plotted based on the results in DAVIS-2016 dataset. The threshold θ_c varies from 0.5 to 1.0 for decision module. The sampling percentage for box propagation varies from 0 to 100% for random assignment. In the periodic assignment, the frames are chosen at every [2,11] fps to be processed with the box re-identification stream.

sequences are divided into 30 ones for training and the other 20 ones for validation. In the DAVIS 2016 dataset, only one foreground object is annotated for each video. The DAVIS 2017 dataset augments the DAVIS 2016 dataset by adding another 30 and 10 sequences to the training and validation datasets, respectively. Unlike DAVIS 2016 dataset, more than one objects are annotated with pixel-level masks in each video of the DAVIS 2017. In total, 10459 frames with 376 object instances are annotated in the DAVIS 2017.

Metrics. To evaluate our model as well as other state-of-the-art approaches, we exploit three metrics including the mean region similarity ($\mathcal{J}$ Mean), mean contour accuracy ($\mathcal{F}$ Mean) and their average ($\mathcal{G}$ Mean) as [30]. Besides, we also provide the run time of each method for efficiency evaluation. The results of other methods are obtained from their published reports or codes. All the experiments are conducted on one NVIDIA 1080Ti GPU.

4.2. Comparison with State-of-the-arts

DAVIS 2016. We compare the performance of our method with state-of-the-art approaches on DAVIS 2016 dataset [29]. In Tab. 1, we list some common operations remains in the existing methods, including online finetuning (OF) and post-processing (PP) (*e.g.,* CRF [21]). In semi-supervised VOS, existing methods usually exploit time-consuming online finetuning or CRF [21] to improve the accuracy of segmentation. For a fair comparison in algorithm speed, the run time of OF and PP is also included. According to the quantitative results as well as run time in Tab. 1, our proposed model achieve the fastest speed with comparable accuracy against state-of-the-arts.

Compared with the existing efficient methods without online finetuning, our model outperforms the best-performed RGMP [34] by 2.5%, 2.2% on $\mathcal{J}$ Mean and $\mathcal{F}$-Mean, respectively. Besides, the computation speed of our model is $2\times$ faster against RGMP. For the methods with online finetuning, our model is much more efficient and achieves similar performance.

Table 3: Ablation study of each module on DAVIS-2016 validation dataset.

Method	$\mathcal{J}$ Mean	$\mathcal{F}$ Mean	$\mathcal{G}$ Mean
w/o FT	67.1	66.5	66.8
w/o PT	68.4	69.9	69.2
w/o DTN	72.7	71.8	72.3
BR with ori RPN	81.7	81.6	81.6
MRN w/o mask guidence	82.1	81.7	81.9
Ours	**83.7**	**83.5**	**83.6**

Table 4: The analysis on the run time of each component in our model. The reported time is tested with DAVIS-2016 dataset on one NVIDIA 1080Ti GPU.

Module	Backbone	DM	BP	BR	MRN
Time	0.021s	0.003s	0.003s	0.071s	0.015s

DAVIS 2017. We also conduct the comparison experiments on DAVIS 2017 [30] dataset to verify the effectiveness of our method on multi-object segmentation. Tab. 2 shows the quantitative comparisons with six state-of-the-arts on three metrics. These results demonstrate that our model is able to achieve better performance than other approaches on DAVIS 2017 dataset.

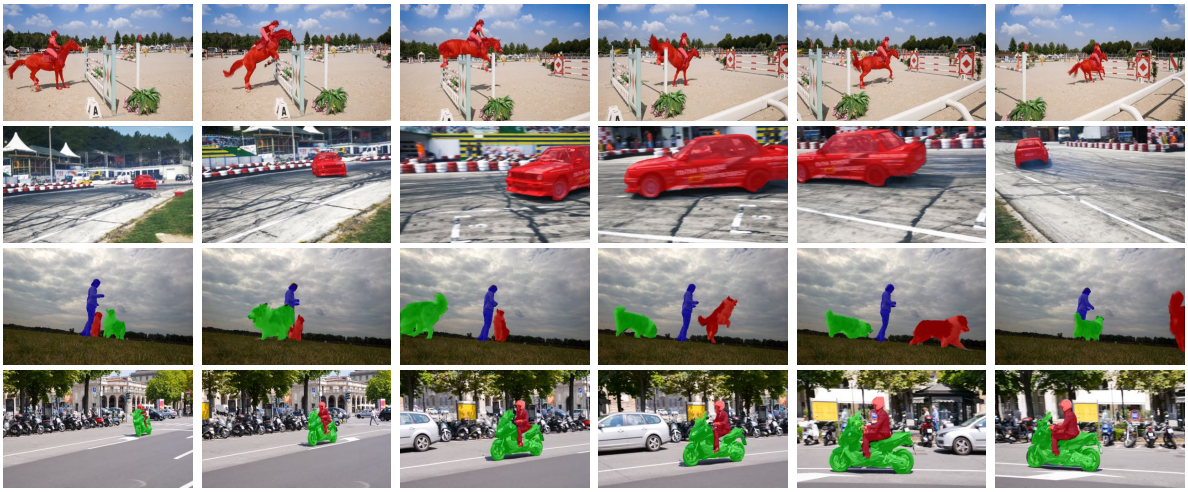
Qualitative results. We illustrate the visual results of our model on DAVIS 2016 and 2017 datasets in Fig. 7. The qualitative results demonstrate that our model can not only consistently track the target objects but also produce accurate segmentation masks with well-defined details.

4.3. Ablation Study

In this section, we analyze the contribution of each component in our model, including dynamic targeting network (DTN), mask refinement network (MRN). The results on DAVIS 2016 dataset are shown in Tab. 3.

Effectiveness of dynamic targeting network. In our model, we employ the DTN to generate a zoomed-in target RoI and produce an object mask within it. To demonstrate the effectiveness of DTN, we feed the raw multi-level features without RoI Align to MRN for generating segmentation (namely "w/o DTN"). The quantitative results in Tab. 3 verify the efficacy of DTN on producing more accurate localization and segmentation. Besides, to verify the effect of confidence score in decision module, we add a comparison model named "Random Assignment", in which the input frames are randomly sampled by various percentages to go through the box propagation stream. For a fair comparison, we report the average result of four various random sampling. The results in Fig. 6 demonstrate the efficacy of DM on assigning proper sub-streams for various frames.

Effectiveness of conditional RPN. We propose a conditional RPN (CRPN), in which the proposals in the current frame are generated based on the location, scale and aspect ratio of previous target RoI. To compare with the original



(a) (b) (c) (d) (e) (f)

Figure 7: Qualitative results of our proposed model on DAVIS 2016 and DAVIS 2017. From left to right: (a) initial frame with user annotation, (b)-(f) segmentation results of subsequent frames.

RPN [11], we sort all the proposals (totally $h_i \times w_i \times 9$) according to their objectness scores and choose the top-100 ones as box candidates for re-identification. We name this model as "BR with ori RPN". Its comparison result with our CRPN is shown in Tab. 3, which proves that our CRPN is able to facilitate more efficient computational speed as well as more precise RoIs for segmentation.

Effectiveness of mask refinement network. In MRN, we use the warped mask from the previous frame as a prior to guide the mask generation in the current frame. To testify its effectiveness, we remove the mask guidance in MRM and name this module as "MRN w/o mask guidence". The comparison results in Tab. 3 demonstrate the contribution of mask guidance in our MRN.

Analysis on training strategy. For training our proposed model, we conduct a two-stage training strategy, including pre-training on instance object segmentation task and finetuning on video object segmentation datasets. In Tab. 3, we report the results of the model skipped the pre-training stage (termed as "w/o PT") and the model skipped the finetuning stage (named as "w/o FT"). The results verify that both training stages contribute to the generation of accurate segmentation in our model.

Analysis of speed vs accuracy. In the DTN, a threshold is pre-defined for determining the model's trade-off between segmentation accuracy and computational efficiency. To verify the influence of thresholds on the overall performance of our model, we report the accuracy ($\mathcal{G}$ Mean) versus frame rate (fps) under various thresholds. The data points on each curve indicate different values of threshold θ_t. It can be observed that as θ_t increases, the data points move to the upper-left corner. This indicates that the segmentation performance increases but the fps decreased. On the contrary, when θ_t decreases, the data points move toward the opposite bottom-right corner, which means that more frame regions pass through the faster box propagation stream. By adjusting the threshold, our proposed model can be customized to meet various requirement for accuracy and efficiency.

Run time analysis. Our model is able to achieve accurate performance with a high speed of 14 fps. In Tab. 4, we analyse the run time of each component in our model, including backbone network, dynamic targeting network, box propagation stream, box re-identification stream and mask refinement network. From the results on run time, we can observe that the box propagation stream is much more efficient than box re-identification stream.

5. Conclusion

We propose a dynamic targeting network and a mask refinement network for video object segmentation. The dynamic targeting network contains two sub-streams, box propagation stream and box re-identification stream, for producing target RoI. The former is faster but less effective at objects with large deformation or occlusion. The latter performs favorable in difficult scenarios but is slower. We utilize a decision module to adaptively determine the sub-stream for each frame. Finally, the segmentation mask of each target RoI is generated by mask refinement network. The experimental results on two benchmarks demonstrate that our model performs favorable against the state-of-the-arts and achieves the highest speed of 14fps.

Acknowledgements. This paper is supported in part by National Natural Science Foundation of China No.61725202, 61829102, 61751212, in part by the Fundamental Research Funds for the Central Universities under Grant No. DUT19GJ201 and gifts from Adobe.

References

[1] Linchao Bao, Baoyuan Wu, and Wei Liu. Cnn in mrf: Video object segmentation via inference in a cnn-based higher-order spatio-temporal mrf. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 5977–5986, 2018.

[2] Sergi Caelles, Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. One-shot video object segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[3] Jingchun Cheng, Yi-Hsuan Tsai, Wei-Chih Hung, Shengjin Wang, and Ming-Hsuan Yang. Fast and accurate online video object segmentation via tracking parts. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[4] Jingchun Cheng, Yi-Hsuan Tsai, Shengjin Wang, and Ming-Hsuan Yang. Segflow: Joint learning for video object segmentation and optical flow. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[5] Hai Ci, Chunyu Wang, and Yizhou Wang. Video object segmentation by learning location-sensitive embeddings. In *Proceedings of European Conference on Computer Vision*, 2018.

[6] Henghui Ding, Xudong Jiang, Ai Qun Liu, Nadia Magnenat Thalmann, and Gang Wang. Boundary-aware feature propagation for scene segmentation. In *Proceedings of the International Conference on Computer Vision*, 2019.

[7] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Context contrasted feature and gated multi-scale aggregation for scene segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2393–2402, 2018.

[8] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Semantic correlation promoted shape-variant context for segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8885–8894, 2019.

[9] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

[10] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[11] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(99):1–1, 2017.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[13] Ping Hu, Gang Wang, Xiangfei Kong, Jason Kuen, and Yap-Peng Tan. Motion-guided cascaded refinement network for video object segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[14] Yuan-Ting Hu, Jia-Bin Huang, and Alexander Schwing. Maskrnn: Instance level video object segmentation. In *Advances in Neural Information Processing Systems*, 2017.

[15] Yuan-Ting Hu, Jia-Bin Huang, and Alexander G. Schwing. Unsupervised video object segmentation using motion saliency-guided spatio-temporal propagation. In *Proceedings of European Conference on Computer Vision*, 2018.

[16] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[17] Varun Jampani, Raghudeep Gadde, and Peter V Gehler. Video propagation networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[18] Won-Dong Jang, Chulwoo Lee, and Chang-Su Kim. Primary object segmentation in videos via alternate convex optimization of foreground and background distributions. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[19] Margret Keuper, Bjoern Andres, and Thomas Brox. Motion trajectory segmentation via minimum cost multicuts. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[20] Anna Khoreva, Rodrigo Benenson, Eddy Ilg, Thomas Brox, and Bernt Schiele. Lucid data dreaming for object tracking. *CoRR*, abs/1703.09554, 2017.

[21] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in Neural Information Processing Systems*, pages 109–117, 2011.

[22] Siyang Li, Bryan Seybold, Alexey Vorobyov, Alireza Fathi, Qin Huang, and C-C Jay Kuo. Instance embedding transfer to unsupervised video object segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[23] Xiaoxiao Li and Chen Change Loy. Video object segmentation with joint re-identification and attention-aware mask propagation. In *Proceedings of European Conference on Computer Vision*, 2018.

[24] Yule Li, Jianping Shi, and Dahua Lin. Low-latency video semantic segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[25] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proceedings of European Conference on Computer Vision*, 2014.

[26] Jonathon Luiten, Paul Voigtlaender, and Bastian Leibe. Premvos: Proposal-generation, refinement and merging for video object segmentation. In *Asian Conference on Computer Vision*, 2018.

[27] Nicolas Maerki, Federico Perazzi, Oliver Wang, and Alexander Sorkine-Hornung. Bilateral space video segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[28] Federico Perazzi, Anna Khoreva, Rodrigo Benenson, Bernt Schiele, and Alexander Sorkine-Hornung. Learning video

object segmentation from static images. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[29] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[30] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alexander Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv:1704.00675*, 2017.

[31] Jae Shin Yoon, Francois Rameau, Junsik Kim, Seokju Lee, Seunghak Shin, and In So Kweon. Pixel-level matching for video object segmentation using convolutional neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017.

[32] Paul Voigtlaender and Bastian Leibe. Online adaptation of convolutional neural networks for video object segmentation. In *Proceedings of British Machine Vision Conference*, 2017.

[33] Wenguan Wang, Jianbing Shen, and Fatih Porikli. Saliency-aware geodesic video object segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[34] Seoung Wug Oh, Joon-Young Lee, Kalyan Sunkavalli, and Seon Joo Kim. Fast video object segmentation by reference-guided mask propagation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[35] Huaxin Xiao, Jiashi Feng, Guosheng Lin, Yu Liu, and Maojun Zhang. Monet: Deep motion exploitation for video object segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[36] Jifeng Dai Lu Yuan Yichen Wei Xizhou Zhu, Yuwen Xiong. Deep feature flow for video recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[37] Yu-Syuan Xu, Tsu-Jui Fu, Hsuan-Kung Yang, and Chun-Yi Lee. Dynamic video segmentation network. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[38] Linjie Yang, Yanran Wang, Xuehan Xiong, Jianchao Yang, and Aggelos K. Katsaggelos. Efficient video object segmentation via network modulation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[39] Lu Zhang, Ju Dai, Huchuan Lu, You He, and Gang Wang. A bi-directional message passing model for salient object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1741–1750, 2018.

[40] Lu Zhang, Jianming Zhang, Zhe Lin, Huchuan Lu, and You He. Capsal: Leveraging captioning to boost semantics for salient object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6024–6033, 2019.