Learning to Recommend Frame for Interactive Video Object Segmentation in the Wild

Zhaoyuan Yin[1], Jia Zheng[2], Weixin Luo[3], Shenhan Qian[4], Hanling Zhang[5]*, Shenghua Gao[4,6]
[1]College of Computer Science and Electronic Engineering, Hunan University
[2]KooLab, Manycore [3]Meituan Group
[4]ShanghaiTech University [5]School of Design, Hunan University
[6]Shanghai Engineering Research Center of Intelligent Vision and Imaging
{zyyin, jh_hlzhang}@hnu.edu.cn jiajia@qunhemail.com luoweixin@meituan.com
{qianshh, gaoshh}@shanghaitech.edu.cn

Abstract

This paper proposes a framework for the interactive video object segmentation (VOS) in the wild where users can choose some frames for annotations iteratively. Then, based on the user annotations, a segmentation algorithm refines the masks. The previous interactive VOS paradigm selects the frame with some worst evaluation metric, and the ground truth is required for calculating the evaluation metric, which is impractical in the testing phase. In contrast, in this paper, we advocate that the frame with the worst evaluation metric may not be exactly the most valuable frame that leads to the most performance improvement across the video. Thus, we formulate the frame selection problem in the interactive VOS as a Markov Decision Process, where an agent is learned to recommend the frame under a deep reinforcement learning framework. The learned agent can automatically determine the most valuable frame, making the interactive setting more practical in the wild. Experimental results on the public datasets show the effectiveness of our learned agent without any changes to the underlying VOS algorithms. Our data, code, and models are available at https://github.com/svip-lab/IVOS-W.

1. Introduction

Video object segmentation aims to segment the objects of interest in a video sequence. It has been widely applied to many downstream applications such as video editing and object tracking. Recently, DAVIS dataset [24, 25] and YouTube-VOS dataset [36] are introduced and significantly drive forward this task. However, collecting such densely-annotated datasets is expensive and time-consuming. For example, labeling a single object in one frame of DAVIS dataset requires more than 100 seconds [2], finally resulting in either limited sizes [24, 25] or coarse annotations [36] in the existing VOS datasets.

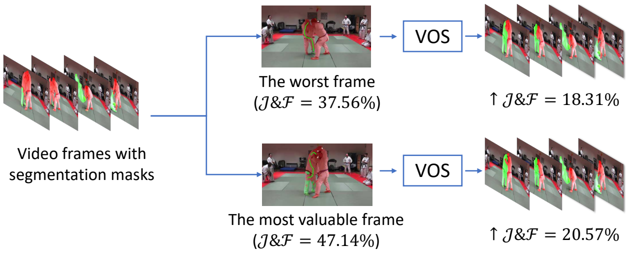

VOS
The worst frame
($\mathcal{J}\&\mathcal{F}$ = 37.56%)
↑ $\mathcal{J}\&\mathcal{F}$ = 18.31%
Video frames with
segmentation masks
VOS
The most valuable frame
($\mathcal{J}\&\mathcal{F}$ = 47.14%)
↑ $\mathcal{J}\&\mathcal{F}$ = 20.57%

Figure 1. The frame with the worst segmentation quality *vs*. the most valuable frame in a single round. The frame with the worst segmentation quality only improves the performance of VOS by 18.31 %, while the most valuable one improves the performance by 20.57 %.

To minimize the human efforts, Caelles et al. [2] introduces a human-in-the-loop VOS setting, *i.e*., the interactive VOS with scribble supervision. Specifically, the interactive VOS algorithm will predict an initial segmentation mask for each frame based on the initial scribbles provided by a user. It will then gradually refine the segmentation masks with additional scribbles of some frames selected by the user, who may evaluate the result by the segmentation quality between the predictions and the ground truths. Whereas, the ground-truth segmentation masks are not available in practice, so the user cannot select a potential frame based on the segmentation quality. Further, the frame with the worst segmentation quality may not be exactly the most valuable one contributing the most to the refinement performance, as shown in Figure 1. In this paper, we claim that *the most valuable frame is not necessarily the one with the worst segmentation quality for the interactive VOS task.*

*Corresponding author.

This paper aims to develop a criterion for determining the *worthiness* of the frame. The worthiness of a frame reflects how much it can improve the segmentation performance across the video sequence if it is selected to provide additional scribbles. However, measuring the worthiness is difficult due to the complexity and variety of videos and uncertainties in the refinement process. To this end, we formulate the frame recommendation problem as a Markov Decision Process (MDP) and train the recommendation agent with Deep Reinforcement Learning (DRL). To narrow the state space, we define the state as the segmentation quality of each frame instead of the image frames and segmentation masks. We also include the recommendation history of each frame in the state. Inspired by Mask Scoring R-CNN [15], we use a segmentation quality assessment module to estimate the segmentation quality. Given the user scribbles on the recommended frame, we leverage the off-the-shelf interactive VOS algorithms [12, 19, 21] to refine the segmentation masks. Without any ground-truth information, the learned agent can recommend the frame. To further evaluate the ability of generalization of our agent, we follow the DAVIS dataset [2] to extend a subset of YouTube-VOS dataset [36] with initial scribbles. The experimental results show that our learned agent outperforms all baseline frame selection strategies on DAVIS and YouTube-VOS dataset without any changes to underlying VOS algorithms, whether the ground truth is available or not.

In summary, **our contributions** are as follows: (i) We demonstrate that the frame used in the current interactive VOS paradigm, *i.e*., the frame with the worst segmentation quality, for user annotation is not the best one. (ii) We propose a novel deep reinforcement recommendation agent for interactive VOS, where the agent recommends the most valuable frame for user annotation. The agent does not require any ground-truth information in the testing phase, therefore it is more practical. (iii) Following the interactive VOS setting [2], we extend a subset of YouTube-VOS dataset [36] with initial scribbles for performance evaluation. (iv) Extensive experiments on the challenging datasets, namely DAVIS dataset and YouTube-VOS dataset, validate the effectiveness of our proposed method.

2. Related Work

2.1. Semi-supervised VOS

Semi-supervised VOS aims to segment objects based on the object mask given in the first frame. With the advent of deep learning in computer vision, Convolutional Neural Networks (CNNs) have recently been investigated to solve the VOS task. One line of work [1, 5, 14, 30, 34, 37] detects the objects using the appearance in the first frame. For instance, OSVOS [1] fine-tunes the network using the first-frame ground truth when testing. FEELVOS [34] uses pixel-level embedding together with a global and local matching mechanism. Another line of work [16, 20, 23] learns to propagate the segmentation mask from one frame to the next. DyeNet [18] takes advantage of both detection and mask propagation approaches. Recently, Griffin and Corso [8] demonstrates that instead of using the first frame as the prior, selecting another frame for annotation will lead to performance improvement. Similar to [8], in this paper, we find this is also applicable to the interactive VOS setting.

2.2. Interactive VOS

Interactive VOS relies on the user input, such as scribbles [2, 12, 19, 21] or points [5], to segment objects of interest in an interactive manner. Caelles et al. [2] proposes a CNN-based method built upon OSVOS and fine-tunes the model based on the user annotations in each round. IPN [21] and ATNet [12] use two segmentation networks to handle interaction and propagation, respectively. ATNet [12] further uses a global and local transfer module to transfer segmentation information to other frames. Built upon FEELVOS, MANet [19] employs a memory aggregation mechanism to record all the previous user annotations. However, all these approaches follow the paradigm of [2] and assume the user selects the frame with the worst segmentation quality. In this paper, we argue that *the frame with the worst segmentation quality is not exactly the one with the most potential performance improvement.*

2.3. Reinforcement Learning in Vision and Video

Reinforcement Learning (RL) is a promising approach to tackle sequential decision-making problems. Many methods try to formulate vision tasks in the spatial and temporal domain as sequential decision-making problems and apply RL to solve them at different levels. Song et al. [31] proposes an RL-based method to gradually generate a set of points for the interactive image segmentation. The generated points are used to refine segmentation via an off-the-shelf segmentation algorithm. Some approaches [4, 27, 28] introduce RL to tackle the object tracking problem by learning to transfer the bounding box of the target object from the previous frame to an appropriate place in a new frame without scanning all the possible regions. Some other approaches are proposed to locate key-frames in a video sequence for more effective processing at the frame-level. For example, Tang et al. [33] use RL to find a criterion to select a set of representative frames for action recognition. Wang et al. [35] learn to locate the activity in a video according to the query language by leveraging an agent to observe selected frames in a video to find the temporal boundaries. Gao et al. [6] locate the start frame of action in an untrimmed video by conducting a class-agnostic start detector based on observing the action scores for each frame. Hu et al. [13] leverage RL to determine a set of most in-

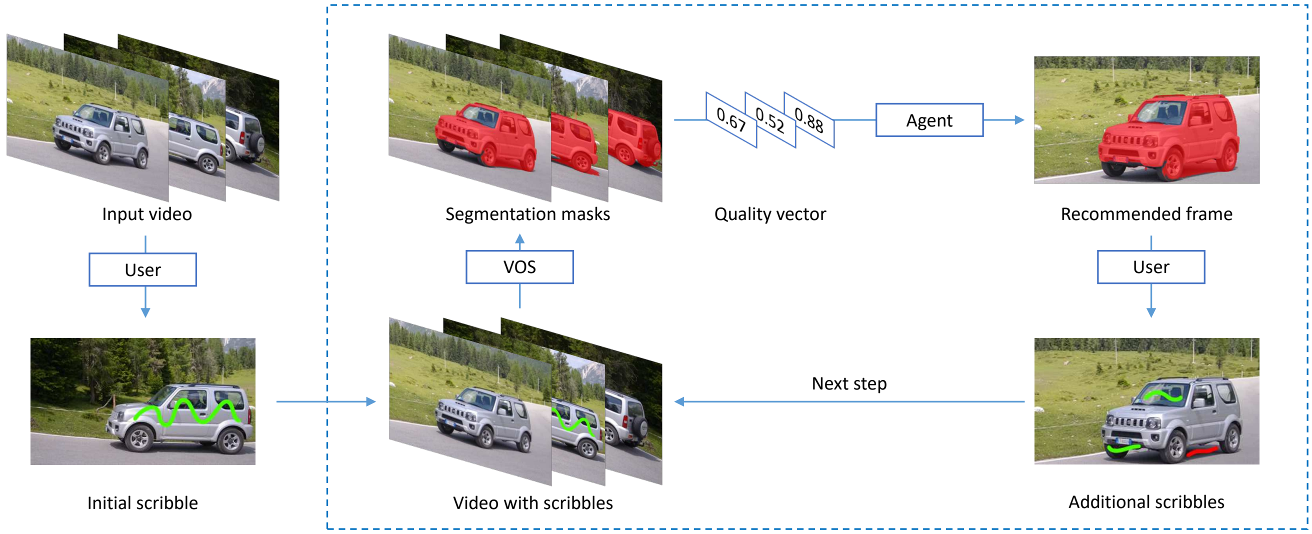


Figure 2. Our proposed interactive VOS framework. In the beginning, the user selects one frame that best represents objects of interest and labels them with initial scribbles. Then, we adopt off-the-shelf VOS algorithm to segment the target object and estimate the segmentation quality for each frame. Afterward, taking the segmentation quality as input, the agent recommends the most valuable frame to the user, who finally draws additional scribbles to refine the masks. Later, the VOS algorithm, the agent, and the user constitute a loop that iteratively refines the predicted masks.

formative frames and group relevant relations inferred from the selected frames.

Recently, RL is also introduced to the VOS. Han et al. [9] integrates the RL into the VOS task to refine the bounding box when propagating the results from the previous frame to the current frame. Gowda et al. [7] group the object proposals sequentially over both space and time. Chai [3] uses RL to locate a patch of the area as hard attention for VOS to perform the segmentation based on a set of collected memory. Sun et al. [32] propose a method to generate a pixel-level region of interest for more effective online adaptation. Similar to [32], our proposed method is based on existing VOS methods but focuses on frame-level optimization rather than pixel-level, which is more compatible with the interactive VOS setting.

3. Methods

Given N frames $\{I^1, I^2, \ldots, I^N\}$, the corresponding segmentation masks $\{M^1, M^2, \ldots, M^N\}$ for the target objects of interest and any previously provided annotations, the agent recommends the most valuable frame for additional user annotation. Figure 2 shows the overall pipeline of our proposed framework.

3.1. Learning to Recommend

We formulate the frame recommendation problem in the interactive VOS as an MDP, where the frame selection is only based on the segmentation masks at each step. Specifically, considering the t-th iteration, the agent observes the segmentation masks, which are regarded as state s_t, and determines the recommendation action a_t. Then, the state s_t is transferred to s_{t+1} by the VOS algorithm, and the corresponding reward r_t will be obtained.

State. Intuitively, the state should contain enough information, such as video frames and segmentation masks. However, this leads to higher dimensional state space. Thus, we use the segmentation quality $q_t \in [0,1]^N$ as a proxy of video frames and masks, where N represents the total number of frames in the video sequence. We further include recommendation history $h_t \in \{0, 1, ..., T\}^N$, where T is the maximum number of interaction, and the n-th value in h_t represents the number of times that the n-th frame is recommended. Thus, the state s_t is defined as:

$$s_t = \textsc{Concat}(q_t, h_t), \tag{1}$$

where $\textsc{Concat}(\cdot)$ denotes the concatenation operator.

Action. The action $a_t \in \{1, \ldots, N\}$ at t-th iteration is to determine the next frame for user annotation. We design a Bi-Directional Long Short-Term Memory (LSTM) based network[1] to learn the expected recommendation agent. The network takes the state s_t as input, and outputs the action a_t. The action of the agent is the recommended frame index.

Reward. The reward reflects the quality of the learned frame recommendation strategy. In the interactive VOS, it is impractical to measure the worthiness of each frame in

[1] Please refer to supplementary material for the detail of network architecture.

a single interaction since the contribution of the annotated frame to the final performance cannot be determined without global optimization. Inspired by [26], we design a goal-only reward based on the final performance P achieved by the action sequence until the maximum number of iterations T is reached.

We expect that the learned recommendation policy is at least better than the random selection policy when $t = T$. To get the performance $\hat{P}$ of the random selection policies, we first run experiments 30 times with the random selection strategy for each training video sequence. We assume that $\hat{P}$ follows the t-distribution, and get the expected mean value $\hat{\mu}$ and variance $\hat{\sigma}$. An intuitive reward function can be designed by comparing the performances between the learned recommendation policy and the random selection policy:

$$r_t^{\text{goal}}(P) = \frac{P - \hat{\mu}}{\hat{\sigma}}. \tag{2}$$

The reward in Eq. (2) is positive when the performance P is greater than the average performance $\hat{\mu}$ of the random selection policy. Otherwise, the reward will be negative.

In practice, we find that it is not sufficient to make learned agent only better than the average performance of the random selection policy. Thus, we set the reward positive only if $P > \hat{\mu} + \hat{\sigma}$. The final reward is formulated as follows:

$$r_t^{\text{goal}}(P) = \frac{P - (\hat{\mu} + \hat{\sigma})}{\hat{\sigma}}. \tag{3}$$

We set the reward $r_t^{\text{goal}} = 0$ when $t < T$, since the contribution of the intermediate actions to the final performance improvement cannot be measured directly.

Due to the motion and viewpoint, the appearance of the object may change significantly in the video. Intuitively, the recommended frames should cover more distinct frames, which may lead to better performance. To this end, we design an auxiliary reward at each step to encourage more diverse recommendation frames and punish the action that is not the fewest one in the action history:

$$r_t^{\text{aux}} = \begin{cases} 1, & a_t = \arg\min h_t, \\ -1, & \text{otherwise}. \end{cases} \tag{4}$$

Double Q-learning. We solve this MDP by the double Q-learning [10]. Considering both two rewards, the underlying action-value function for the step t is defined as follows:

$$Q_t^* = \begin{cases} \delta \cdot r_t^{\text{goal}}, & t = T, \\ \delta \cdot r_t^{\text{aux}} + \gamma \cdot Q^T(s_{t+1}, a_{t+1}), & t < T, \end{cases} \tag{5}$$

we set the scaling factor $\delta = 0.1$ and the discount factor $\gamma = 0.95$. We use the policy network $Q^P(\cdot)$ to determine the action by $a_{t+1} = \text{argmax}_a Q^P(s_{t+1}, a)$, and use the target network $Q^T(\cdot)$ to evaluate the value of the action a_{t+1} by $Q^T(s_{t+1}, a_{t+1})$.

We use the mean squared error loss $\text{MSE}(\cdot)$ to supervise the learning of the agent:

$$\mathcal{L}_{\text{agent}} = \text{MSE}(Q_t, Q_t^*). \tag{6}$$

Task decomposition. The standard RL focuses on maximizing the reward received from the whole episode (*e.g.*, actions across the T interactions) and only consider maximizing the final performance. However, the interactive VOS aims to achieve the highest performance with minimal interactions. This motivates us to treat any interaction as an independent procedure and decompose the frame recommendation task with a maximum number of T iterations into T sub-tasks to maximize the performance at each interaction. For each sub-task, the maximum number of iteration $T' = 1, \dots, T$. Thus, s_t can be intermediate state in the sub-task with $t < T'$. Meanwhile, the s_t is the terminal state for sub-task with $t = T'$.

3.2. Segmentation Quality Assessment

To narrow the state space, we use the segmentation quality as a proxy state for our frame recommendation agent. Inspired by Mask Scoring R-CNN [15], we use a quality assessment module to estimate the segmentation quality for each frame.

Suppose that there are K target objects of interest in the video. We first calculate the tight bounding box $B^{n,k}$ containing the foreground mask for each instance k based on the segmentation probability map $\hat{M}^{n,k}$. Then, we enlarge the bounding box $B^{n,k}$ by a factor of 1.5. To ignore the background regions, we crop the RGB image I^n and corresponding probability map $\hat{M}^{n,k}$ based on the enlarged bounding box $B^{n,k}$. Then, we concatenate the cropped RGB image and probability map as the input of the segmentation quality assessment module and obtain the segmentation quality estimation $q^{n,k}$ of each object of interest. We implement this module with a ResNet-50 [11] followed by a fully connected layer from 2048 to 1.

The segmentation quality q^n of each frame is the average segmentation qualities over all objects of interest within each frame:

$$q^n = \frac{1}{K} \sum_{k=1}^{K} q^{n,k}. \tag{7}$$

We use $\text{MSE}(\cdot)$ to supervise the learning of the segmentation quality:

$$\mathcal{L}_{\text{quality}} = \text{MSE}(q^{n,k}, q^{*,n,k}), \tag{8}$$

where $q^{*,n,k}$ is the corresponding ground-truth segmentation quality.

Setting	Strategy	DAVIS			YouTube-VOS		
		IPN [21]	MANet [19]	ATNet [12]	IPN [21]	MANet [19]	ATNet [12]
Oracle	Worst	48.02	70.85	73.68	**44.67**	66.03	74.89
	Ours	**48.25**	**71.11**	**74.01**	43.86	**66.90**	**75.37**
Wild	Random	47.52(4)	69.81(1)	72.99(3)	43.22(5)	64.97(8)	74.11(8)
	Linspace	46.97	70.10	72.93	42.75	64.75	73.47
	Worst	47.26	69.32	73.33	43.29	65.98	74.69
	Ours	**47.99**	**70.82**	**74.10**	**43.69**	**66.85**	**75.33**

Table 1. Quantitative results (AUC) of the interactive VOS on DAVIS and YouTube-VOS dataset.

3.3. Training and Inference

Training. We train the frame recommendation agent on DAVIS dataset. We adopt the VOS algorithm, *i.e.*, ATNet [12], as the state transition function. It is impractical to train the agent using the whole video sequence due to the various sequence lengths. For each original training sequence, we sample N' consecutive frames to form the subsequence and use the corresponding ground-truth segmentation quality to form the state. We use the experience replay mechanism [29] to make the training process more stable. At the beginning of the agent training, we fill the experience buffer by randomly selecting the frames and then train the agent with ϵ-greedy policy. To train the segmentation quality assessment module, we use the segmentation masks generated by ATNet.

Inference. Given a test video sequence and initial segmentation masks, the segmentation quality assessment module first estimates the segmentation quality for each frame. Then, the agent takes the segmentation quality and recommendation history as input and outputs Q value for each frame. Finally, we recommend the frame with the highest Q value for user annotation. During testing, we use the whole video sequence.

4. Experiments

In this section, we conduct experiments to evaluate the performance of the proposed method on DAVIS dataset [25] and YouTube-VOS dataset [36].

4.1. Dataset and Evaluation Metrics

Datasets. DAVIS dataset [25] contains 60 training sequences and 30 validate sequences. DAVIS dataset provides high-quality densely-annotated segmentation mask annotation for each frame. To test the generalization of the proposed method, we further sample 50 sequences from YouTube-VOS dataset [36]. Since YouTube-VOS dataset does not contain the applicable annotations for the interactive VOS task, we extend such initial scribbles by following [2].

Evaluation metrics. To validate the performance of segmentation, we use the region similarity in terms of intersection over union $\mathcal{J}$ and the boundary accuracy $\mathcal{F}$ as used in [24]. Caelles et al. [2] propose to use the curve of $\mathcal{J}\&\mathcal{F}$ versus the time. Since we focus on evaluating the frame selection strategy, we do not take the time into account. Instead, we use the curve of $\mathcal{J}\&\mathcal{F}$ versus the number of interactions and measure its Area Under Curve (AUC) to validate the interactive setting. The segmentation quality is also measured by $\mathcal{J}\&\mathcal{F}$.

4.2. Implementation Details

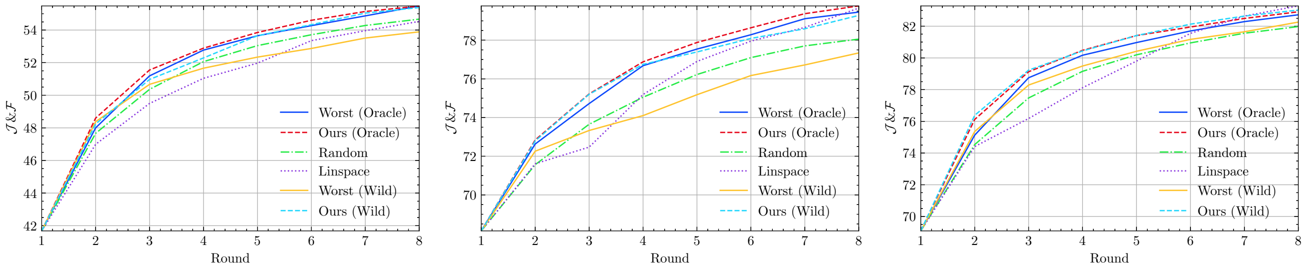
We implement our model with PyTorch [22] and train all models on a single NVIDIA Tesla V100 GPU device. We use Adam [17] optimizer with learning rate 5×10^{-6} and batch size 32. The experience buffer is set to 5760. ϵ decreases from 0.7 to 0.25 over 5000 steps exponentially. To accelerate the training process, we set the maximum of interactions $T = 5$ during training, and $T = 8$ during testing following [2]. It is impractical to generate the scribble annotations by human annotators during training, so we use the human-simulated scribbles[2] by comparing the segmentation predictions and corresponding ground truths. We set $N' = 25$ to the length of the shortest sequence in the training set. It takes approximately 10 hours for the agent to converge.

4.3. Main Results

Strategies for comparison. We compare our learned agent under two settings:

- **"Oracle"**: When the ground-truth segmentation mask is available, we compare our method with [2], *i.e.*, select the frame with the worst segmentation quality ("Worst"). In this setting, we feed the ground-truth segmentation quality to our agent.
- **"Wild"**: When the ground-truth segmentation mask is unavailable, we compare our agent with the following

[2] https://github.com/albertomontesg/davis-interactive



(a) IPN [21] (b) MANet [19] (c) ATNet [12]

Figure 3. The curve of $\mathcal{J}\&\mathcal{F}$ versus the number of rounds on DAVIS dataset.

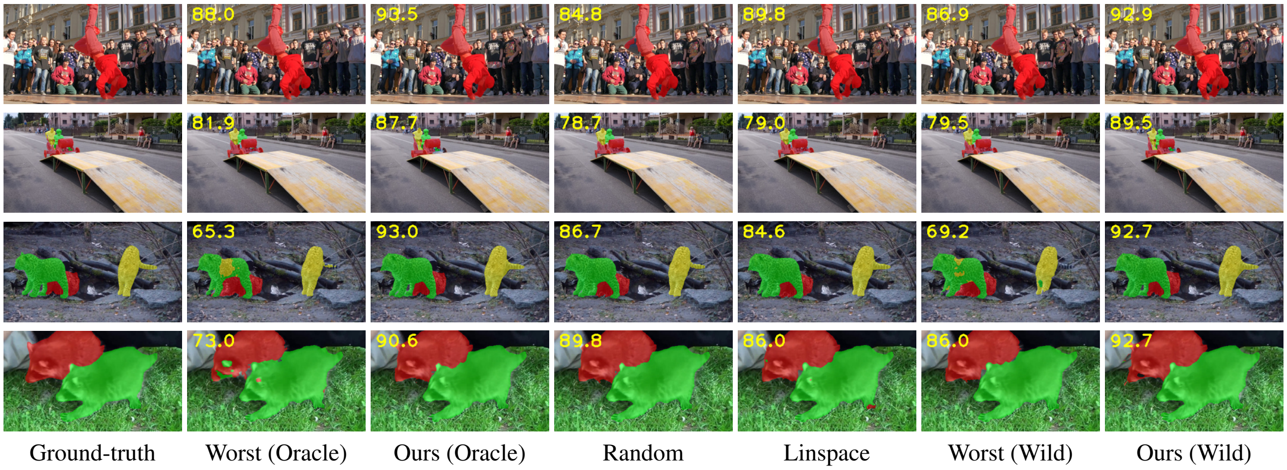


Figure 4. Qualitative comparison on DAVIS (the first two rows) and YouTube-VOS dataset (the last two rows). All result masks are sampled after 8 rounds. The ground truth is available ("Oracle") in the second and third columns, while the ground truth is unknown ("Wild") in the last four columns. We show the segmentation quality $\mathcal{J}\&\mathcal{F}$ on each frame.

frame selection strategies: (i) select uniformly from all frames ("Random"), (ii) select frames with a fixed-length step ("Linspace"), (iii) "Worst". In this setting, we use the predicted segmentation quality for "Worst". We run "Random" selection strategy 5 times and report the mean and variance.

Segmentation algorithm. We choose three off-the-shelf interactive VOS approaches, IPN [21][3], MANet [19][4] and ATNet [12][5], based on their performance and source code availability. All the segmentation algorithms are only trained on DAVIS dataset.

Quantitative evaluation. Table 1 shows the quantitative results on DAVIS dataset and YouTube-VOS dataset. We make the following observations: (i) Our learned agent achieves state-of-the-art performance on DAVIS dataset and generalizes well to YouTube-VOS dataset without any changes to the underlying VOS algorithms, no matter if the ground truth is available or not. (ii) Our agent outperforms the worst frame selection strategy (with the exception of IPN) when ground truth is available ("Oracle"), demonstrating that the frame with the worst evaluation result is not exactly the best one for user annotation. (iii) When ground truth is not available ("Wild"), our method also outperforms all baseline strategies. Due to the space limitation, we refer readers to the supplementary material for the curves of all results on YouTube-VOS dataset.

Figure 3 shows the curves of $\mathcal{J}\&\mathcal{F}$ versus the number of rounds on DAVIS dataset. As one can see, our agent outperforms other frame selection strategies in all rounds when ground truth is available ("Oracle"). When ground truth is not available ("Wild"), our agent can outperform all baselines, *i.e.*, Random, Worst, and Linspace.

Qualitative evaluation. Figure 4 shows the qualitative results of ATNet on DAVIS validation set. We sample results generated by the different frame selection policies after 8 rounds. As one can see, our approach produces accurate segmentation masks. We also show the frames recommended by our agent and the worst frames at each round in Figure 5. The worst frame selection strategy tends to select a small range of frames. However, the user could not pro-

[3] https://github.com/seoungwugoh/ivs-demo

[4] https://github.com/lightas/CVPR2020_MANet

[5] https://github.com/yuk6heo/IVOS-ATNet

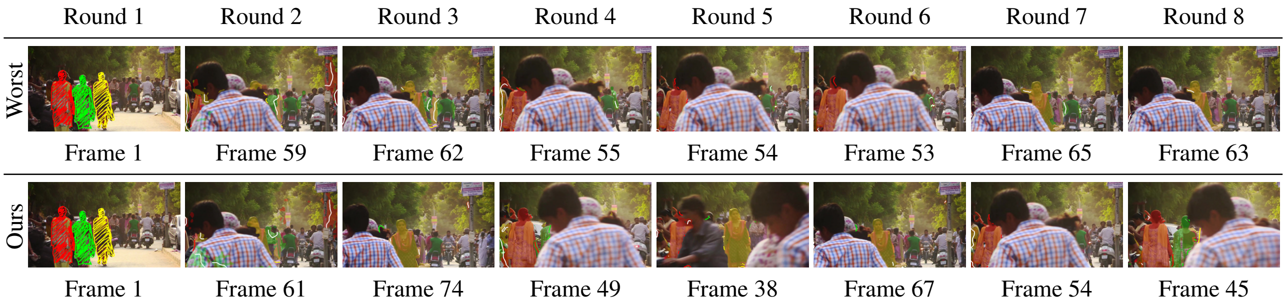


Figure 5. Recommended frames for the "india" sequence in DAVIS dataset. The worst frame selection strategy in the top row achieves $66.92\,\%$ in terms of $\mathcal{J}\&\mathcal{F}$, while ours in the bottom row achieves $72.25\,\%$.

Annotator	AUC	Time (s)
Human	73.09	14.01
Ours	**74.10**	**0.70**

Table 2. Comparison with humans on DAVIS dataset.

VOS	PCC	MSE
IPN [21]	0.47	0.05
MANet [19]	0.42	0.01
ATNet [12]	0.51	0.01

Table 3. Quantitative results of the segmentation quality assessment module on DAVIS dataset.

vide additional information for the objects on these frames. We refer readers to supplementary materials for more qualitative results.

Comparison with humans. We further compare our proposed frame recommendation agent with the human on DAVIS validation set. In this experiment, we adopt ATNet as the VOS algorithm. We overlay the segmentation mask on the RGB image and show the overlaid frame to the human. We only ask the human to select the valuable frame for annotation, and then the chosen frame is annotated by the human-simulated scribbles [2]. As shown in Table 2, our learned agent outperforms the human in both accuracy and efficiency.

Evaluation of the segmentation quality assessment. To validate the accuracy of the segmentation quality assessment module, we adopt the Pearson correlation coefficient (PCC) and MSE between predictions and their ground truths. As shown in Table 3, the regression model trained only on the data generated by ATNet can also generalize well to other VOS algorithms.

4.4. Ablation Studies

We run several ablation studies to analyze the frame recommendation agent. In all ablation studies, we adopt the ATNet as the VOS algorithm and report the AUC on DAVIS validation set.

Variants	Oracle	Wild
Eq. (2)	71.82	71.97
Eq. (3)	**74.01**	**74.10**

Table 4. Reward function.

Variants	Oracle	Wild
r^{goal}	73.75	73.76
r^{aux}	72.06	72.00
$r^{\text{goal}} + r^{\text{aux}}$	**74.01**	**74.10**

Table 5. Reward.

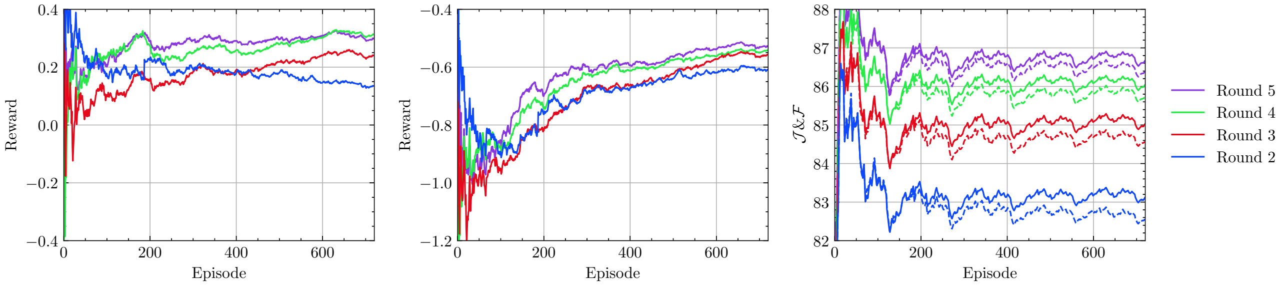
Variants	Oracle	Wild
q_t	73.71	73.55
h_t	73.92	73.92
$q_t + h_t$	**74.01**	**74.10**

Table 6. State.

Reward function. We first verify the effectiveness of the proposed reward function. We compare the two reward functions, *i.e.*, Eq. (2) and Eq. (3). The reward in Eq. (2) is positive if $P > \hat{\mu}$, while the reward in Eq. (3) is positive if $P > \hat{\mu} + \hat{\sigma}$. The results are shown in Table 4. As expected, the proposed reward function has better results.

We further show the change in reward according to the training episode from 2nd round to 5th round in Figure 6. As shown in Figure 6a, the reward in Eq. (2) is mostly positive, and the performance is hard to improve after a certain training episode. In contrast, the reward in Eq. (3) shown in Figure 6b can continuously improve. Figure 6c shows the change in $\mathcal{J}\&\mathcal{F}$ of the training process. As expected, the reward in Eq. (3) has better results than that in Eq. (2).

Reward. To evaluate the effectiveness of goal-only reward r^{goal} in Eq. (3) and auxiliary reward r^{aux} in Eq. (4), we remove either one of them. The results are shown in Table 5. As one can see, the agent trained with both rewards achieves the best performance. This demonstrates that both segmentation quality and frame selection diversity are helpful to the frame recommendation.



(a) The reward curve using Eq. (2).

(b) The reward curve using Eq. (3).

(c) The performance curve.

Figure 6. Training curves on DAVIS dataset. (a) and (b) show the reward obtained in each round with the reward function in Eq. (2) and Eq. (3), respectively. (c) shows the segmentation quality in each round. The dashed and solid lines in (c) represent the performance based on Eq. (2) and Eq. (3), respectively. All curves are smoothed using a weighed moving average algorithm.

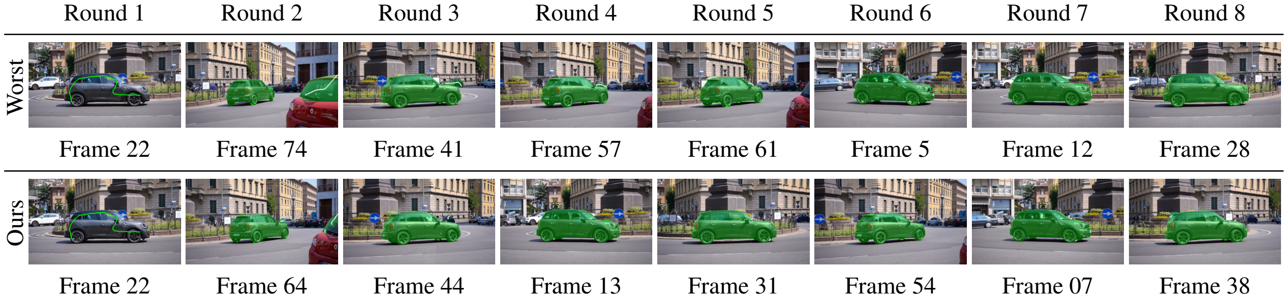


Figure 7. Failure case. We show the recommended frames for the "car-roundabout" sequence in DAVIS dataset. The worst frame selection strategy in the top row achieves 96.75 % in terms of $\mathcal{J}\&\mathcal{F}$, while ours in the bottom row achieves 94.43 %.

Variants	Oracle	Wild
✗	72.58	72.55
✓	**74.01**	**74.10**

Table 7. Task decomposition.

State. To evaluate the effectiveness of the state, we remove either the segmentation quality q_t or the recommendation history h_t. As shown in Table 6, both states play an important role in representing the agent.

Task decomposition. Finally, we investigate the effectiveness of the task decomposition. We train an agent without the task decomposition. The results are shown in Table 7. The agent with task decomposition performs better, which illustrates the advantage of the task decomposition.

4.5. Failure Case

Figure 7 shows the failure case. In this case, the foreground object (*i.e.*, car) moves smoothly away from the camera. As the appearance of the foreground object does not change significantly, the VOS algorithm works very well across the whole video sequence. Thus, it is hard for the agent to select the most valuable frame. Our agent achieves comparable performance to the "Worst" strategy (94.43 % *vs*. 96.75 %).

5. Conclusion

This paper hypothesizes that the frame with the worst segmentation quality selected in the current interactive VOS is not exactly the best one for annotation. To this end, we formulate the frame recommendation problem as a Markov Decision Process and solve it in the DRL manner. The experimental results on public datasets show that our learned recommendation agent outperforms all baseline strategies without any changes to the underlying VOS algorithms.

Acknowledgements

This work was supported by the Special Funds for the Construction of Innovative Provinces in Hunan (2019NK2022), NSFC (61672222, 61932020), National Key R&D Program of China (2018AAA0100704), Science and Technology Commission of Shanghai Municipality (20ZR1436000), and "Shuguang Program" by Shanghai Education Development Foundation and Shanghai Municipal Education Commission. We gratefully acknowledge NVIDIA for GPU donation. We thank the authors of YouTube-VOS for dataset construction.

References

[1] Sergi Caelles, Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. One-shot video object segmentation. In *CVPR*, pages 221–230, 2017. 2

[2] Sergi Caelles, Alberto Montes, Kevis-Kokitsi Maninis, Yuhua Chen, Luc Van Gool, Federico Perazzi, and Jordi Pont-Tuset. The 2018 davis challenge on video object segmentation. *CoRR*, abs/1803.00557, 2018. 1, 2, 5, 7

[3] Yuning Chai. Patchwork: A patch-wise attention network for efficient object detection and segmentation in video streams. In *ICCV*, pages 3415–3424, 2019. 3

[4] Boyu Chen, Dong Wang, Peixia Li, Shuang Wang, and Huchuan Lu. Real-time 'actor-critic' tracking. In *ECCV*, pages 318–334, 2018. 2

[5] Yuhua Chen, Jordi Pont-Tuset, Alberto Montes, and Luc Van Gool. Blazingly fast video object segmentation with pixel-wise metric learning. In *CVPR*, pages 1189–1198, 2018. 2

[6] Mingfei Gao, Mingze Xu, Larry S Davis, Richard Socher, and Caiming Xiong. Startnet: Online detection of action start in untrimmed videos. In *ICCV*, pages 5542–5551, 2019. 2

[7] Shreyank N Gowda, Panagiotis Eustratiadis, Timothy Hospedales, and Laura Sevilla-Lara. Alba: Reinforcement learning for video object segmentation. In *BMVC*, 2020. 3

[8] Brent A. Griffin and Jason J. Corso. Bubblenets: Learning to select the guidance frame in video object segmentation by deep sorting frames. In *CVPR*, pages 8914–8923, 2019. 2

[9] Junwei Han, Le Yang, Dingwen Zhang, Xiaojun Chang, and Xiaodan Liang. Reinforcement cutting-agent learning for video object segmentation. In *CVPR*, pages 9080–9089, 2018. 3

[10] Hado van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *AAAI*, pages 2094–2100, 2016. 4

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 4

[12] Yuk Heo, Yeong Jun Koh, and Chang-Su Kim. Interactive video object segmentation using global and local transfer modules. In *ECCV*, pages 297–313, 2020. 2, 5, 6, 7

[13] Guyue Hu, Bo Cui, Yuan He, and Shan Yu. Progressive relation learning for group activity recognition. In *CVPR*, pages 980–989, 2020. 2

[14] Yuan-Ting Hu, Jia-Bin Huang, and Alexander G Schwing. Videomatch: Matching based video object segmentation. In *ECCV*, pages 54–70, 2018. 2

[15] Zhaojin Huang, Lichao Huang, Yongchao Gong, Chang Huang, and Xinggang Wang. Mask scoring r-cnn. In *CVPR*, pages 6409–6418, 2019. 2, 4

[16] Anna Khoreva, Rodrigo Benenson, Eddy Ilg, Thomas Brox, and Bernt Schiele. Lucid data dreaming for video object segmentation. *IJCV*, 127(9):1175–1197, 2019. 2

[17] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 5

[18] Xiaoxiao Li and Chen Change Loy. Video object segmentation with joint re-identification and attention-aware mask propagation. In *ECCV*, pages 90–105, 2018. 2

[19] Jiaxu Miao, Yunchao Wei, and Yi Yang. Memory aggregation networks for efficient interactive video object segmentation. In *CVPR*, pages 10366–10375, 2020. 2, 5, 6, 7

[20] Seoung Wug Oh, Joon-Young Lee, Kalyan Sunkavalli, and Seon Joo Kim. Fast video object segmentation by reference-guided mask propagation. In *CVPR*, pages 7376–7385, 2018. 2

[21] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Fast user-guided video object segmentation by interaction-and-propagation networks. In *CVPR*, pages 5247–5256, 2019. 2, 5, 6, 7

[22] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NeurIPS Workshop*, 2017. 5

[23] Federico Perazzi, Anna Khoreva, Rodrigo Benenson, Bernt Schiele, and Alexander Sorkine-Hornung. Learning video object segmentation from static images. In *CVPR*, pages 2663–2672, 2017. 2

[24] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, pages 724–732, 2016. 1, 5

[25] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alexander Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *CoRR*, abs/1704.00675, 2017. 1, 5

[26] Chris Reinke, Eiji Uchibe, and Kenji Doya. Average reward optimization with multiple discounting reinforcement learners. In *ICONIP*, pages 789–800, 2017. 4

[27] Liangliang Ren, Jiwen Lu, Zifeng Wang, Qi Tian, and Jie Zhou. Collaborative deep reinforcement learning for multi-object tracking. In *ECCV*, pages 586–602, 2018. 2

[28] Liangliang Ren, Xin Yuan, Jiwen Lu, Ming Yang, and Jie Zhou. Deep reinforcement learning with iterative shift for visual tracking. In *ECCV*, pages 684–700, 2018. 2

[29] Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. In *ICLR*, 2016. 5

[30] 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 *ICCV*, pages 2167–2176, 2017. 2

[31] Gwangmo Song, Heesoo Myeong, and Kyoung Mu Lee. Seednet: Automatic seed generation with deep reinforcement learning for robust interactive segmentation. In *CVPR*, pages 1760–1768, 2018. 2

[32] Mingjie Sun, Jimin Xiao, Eng Gee Lim, Yanchun Xie, and Jiashi Feng. Adaptive roi generation for video object segmentation using reinforcement learning. *Pattern Recognition*, page 107465, 2020. 3

[33] Yansong Tang, Yi Tian, Jiwen Lu, Peiyang Li, and Jie Zhou. Deep progressive reinforcement learning for skeleton-based action recognition. In *CVPR*, pages 5323–5332, 2018. 2

[34] Paul Voigtlaender, Yuning Chai, Florian Schroff, Hartwig Adam, Bastian Leibe, and Liang-Chieh Chen. Feelvos: Fast end-to-end embedding learning for video object segmentation. In *CVPR*, pages 9481–9490, 2019. 2

[35] Weining Wang, Yan Huang, and Liang Wang. Language-driven temporal activity localization: A semantic matching reinforcement learning model. In *CVPR*, pages 334–343,

2019. 2

[36] Ning Xu, Linjie Yang, Yuchen Fan, Jianchao Yang, Dingcheng Yue, Yuchen Liang, Brian Price, Scott Cohen, and Thomas Huang. Youtube-vos: Sequence-to-sequence video object segmentation. In *ECCV*, pages 585–601, 2018. 1, 2, 5

[37] Xiaohui Zeng, Renjie Liao, Li Gu, Yuwen Xiong, Sanja Fidler, and Raquel Urtasun. Dmm-net: Differentiable mask-matching network for video object segmentation. In *ICCV*, pages 3929–3938, 2019. 2