AGSS-VOS: Attention Guided Single-Shot Video Object Segmentation

Huaijia Lin[1] Xiaojuan Qi[2] Jiaya Jia[1,3]
[1]The Chinese University of Hong Kong [2]University of Oxford [3]Tencent YouTu Lab
linhj@cse.cuhk.edu.hk, xiaojuan.qi@eng.ox.ac.uk, leojia@cse.cuhk.edu.hk

Abstract

Most video object segmentation approaches process objects separately. This incurs high computational cost when multiple objects exist. In this paper, we propose AGSS-VOS to segment multiple objects in one feed-forward path via instance-agnostic and instance-specific modules. Information from the two modules is fused via an attention-guided decoder to simultaneously segment all object instances in one path. The whole framework is end-to-end trainable with instance IoU loss. Experimental results on Youtube-VOS and DAVIS-2017 dataset demonstrate that AGSS-VOS achieves competitive results in terms of both accuracy and efficiency.

1. Introduction

Video object segmentation (VOS) aims at segmenting objects in all frames of a video. It finds various applications in video editing, autonomous driving, robotics, human-computer interaction, to name a few. In this paper, we study this problem in the semi-supervised setting, in which annotation of one or multiple objects is given for the first frame in a video. The task is then to segment all corresponding objects in the rest of the video.

Successful approaches [26, 29, 2, 7, 15] on video object segmentation in the semi-supervised setting can be coarsely cast into three categories. One major stream [26, 29, 25] is to separately segment objects and does not consider one-pass multi-object processing. The efficiency is shown in Fig. 1 (in the red curve). Another line of research [15, 11] utilizes region proposals to generate mask proposals. They adopt re-identification networks to find and associate objects. Albeit improving performance, these systems are still time-consuming, *i.e.*, taking 37 seconds per frame [15], and need post-processing to handle false positive object proposals. Recently, embedding-based solutions [2, 7, 17] that measure pixel distance in the embedding space demonstrate great efficiency and accuracy trade-off. These methods have indispensable pixel-wise distance calculation process with $\mathcal{O}(N^2)$ time complexity on $\mathcal{O}(N)$ pixels. They are still difficult to handle high-res videos with memory constraint.

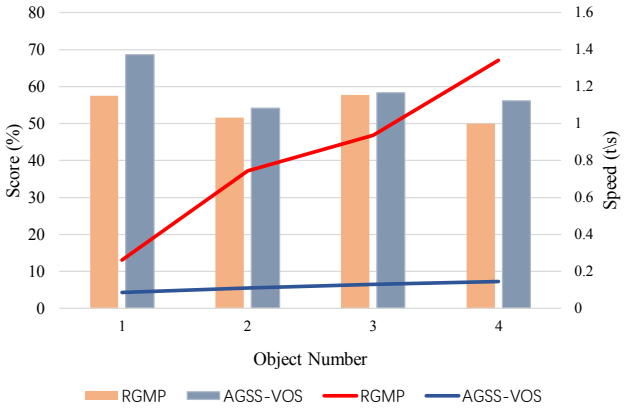

Figure 1. Comparison of accuracy (in histogram) and computation speed (by the curve) regarding different object numbers of RGMP [26] and our method. Input frame size 832×448 is used in DAVIS-2017 test-dev set. On large object numbers, our method efficiency is less affected.

To tackle the above challenges, we propose an end-to-end attention guided single-shot video object segmentation (AGSS-VOS) framework to simultaneously segment all objects in one feed-forward pass without utilizing complex object proposals or time consuming pixel-wise distance calculation. The key idea is to adopt an instance-agnostic module to capture knowledge shared by all instances, and an instance-specific module to generate instance-specific features. Output from the two modules is fused via an attention mechanism to segment object instances.

Specifically, without discriminating among different object instances, the instance-agnostic module takes all objects and encodes them into one common feature with fully convolutional neural networks. The instance-specific module then encodes different objects into separate attention features. The generated two types of features are combined via multiplication and are further utilized to generate masks of instances. Finally, they are normalized to produce the object segmentation prediction for the target frame. The whole framework is end-to-end trainable with instance IoU loss.

Our framework saves computation by processing the reference and target frames involving all objects only once in the instance-agnostic module, while retaining high accuracy via our light-weighted instance-specific component and attention guided decoding scheme. As shown in Fig. 1 (in blue curve), the running time of AGSS-VOS for segmenting multiple objects increases much slower than the single object propagation baseline RGMP [26] as object number increases. Meanwhile, we achieve comparable accuracy.

We experiment with our method on both Youtube-VOS and DAVIS-2017 datasets. Results demonstrate that our method is efficient. The contribution is summarized below.

- We propose an end-to-end attention guided single-shot video object segmentation framework to simultaneously segment multiple objects in one feed-forward path.
- We model instance-specific information as attention features to discriminate among different objects on top of instance-agnostic feature.
- Our approach exhibits high efficiency while retaining reasonable accuracy.

2. Related Work

Existing semi-supervised VOS approaches can be roughly categorized into three directions: 1) single object based VOS where each object instance is separately processed; 2) region proposal based VOS; 3) embedding based VOS.

Single Object VOS In the inference stage, many single-object video object segmentation approaches rely on online learning technique, which needs time-consuming fine tuning on the first annotated frame. OSVSO [1] trained a convolutional network in the training set and adopted online learning in the target video. OnAVOS [24, 23] and OSVOS-S [16] extended OSVOS via an online adaptation mechanism and an instance segmentation network. MaskTrack [18] utilized previous frame mask to guide current segmentation. LucidTracker [10] extended MaskTrack by an extensive data augmentation strategy. LSE [4] proposed location sensitive embedding strategy to refine foreground prediction.

There are offline training approaches without computational expensive online fine-tuning. Yang *et al*. [29] manipulated the intermediate layer of the segmentation network with a modulator to adapt the change of visual and spatial information for each target object. FAVOS [3] utilized a tracking based approach to track bounding boxes for object parts and segmented boxes with a ROI segmentation network. MaskRNN [6] adopted a Mask R-CNN [5] based framework to predict box and corresponding mask for each object. Tokmakov *et al*. [21] and Xu *et al*. [27] proposed convGRU and convLSTM to build a memory module for recursively long-term prediction. AGAM [9] learned a probabilistic generative model of the target and background feature distributions for efficient segmentation.

The most related work to ours is RGMP [26], which proposed a Siamese encoder-decoder network with two-stream input. The reference stream takes the reference frame with the annotated object as input and the target stream takes the target frame with the previous mask as input. The two-stream information is encoded to the same deep feature space and is fused with a global convolution block. The fused feature is further decoded with skip connection from the target stream to produce segmentation for the target frame. RGMP is designed for single-object segmentation while our approach adopts RGMP as an instance-agnostic module to acquire knowledge shared by all instances in one feed-forward path.

Region Proposal Based VOS Approaches along this line adopt region proposal networks (RPN [20]) to generate multiple object proposals shared by all target objects in one feed-forward path. DyeNet [11] has a Re-MP module for object propagation and a Re-ID module on the RPN for associating objects and retrieving missing objects. PReMVOS [15, 13, 14] combined four stream networks, including Mask R-CNN [5] to generate mask proposals, to achieve impressive results with online learning. Although these approaches are able to achieve high accuracy, the dependency on region proposal networks makes it complex in training. Region proposal based approaches typically require post-processing to remove false positive proposals.

Embedding Based VOS Embedding based VOS approaches [2, 7] learn mapping pixels in the reference and target frames in the same embedding space. Different instances are grouped together by comparing feature space distance. FEELVOS [17] extended the method of [2] for multiple object segmentation, which shares the same goal as ours. The whole system can predict multiple objects in one feed-forward path and can be trained in an end-to-end manner. Albeit improving accuracy and efficiency, the approach still suffer from resource issues to handle high-resolution videos due to pixel-pixel embedding.

3. Our Method

Our AGSS-VOS architecture is illustrated in Figure 2. It includes an instance-agnostic module (Figure 2(a)) to extract high-level features shared by all instances, and an instance-specific module (Figure 2(b)) to produce instance-aware feature maps. The two modules are linked via an

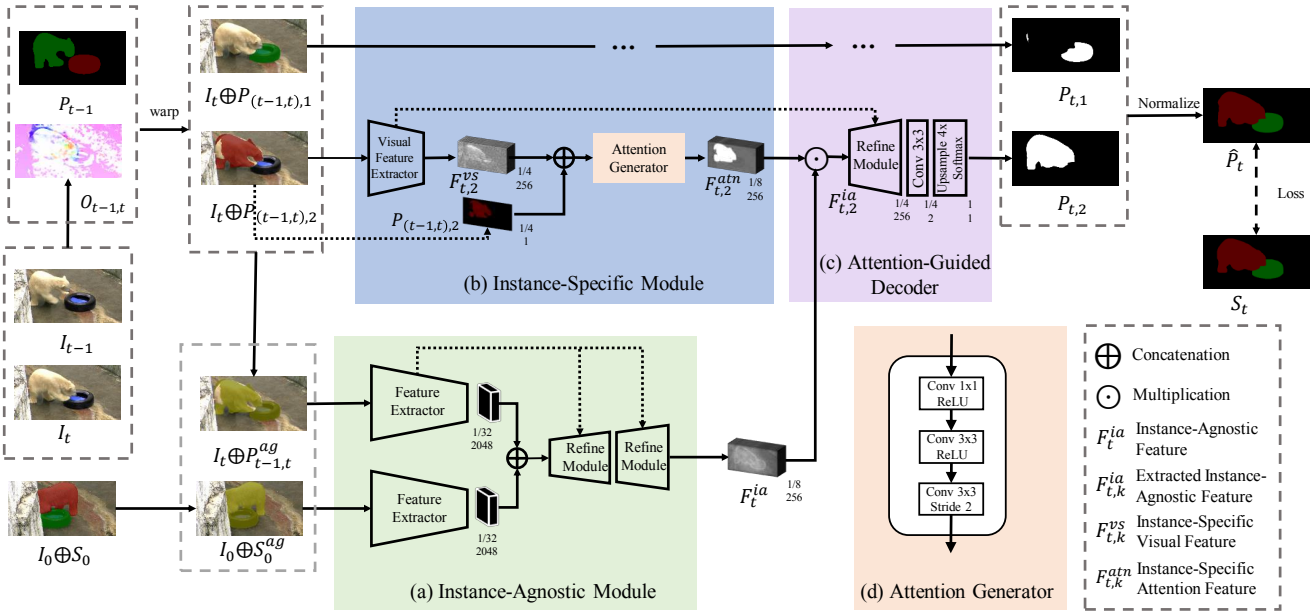

Figure 2. Overview of AGSS-VOS, consisting of (a) an Instance-Agnostic module, (b) an Instance-Specific module and (c) an Attention-Guided decoder. I_t, S_t and P_t denote the image, ground-truth segmentation mask and prediction result in frame t. $O_{t-1,t}$ denotes optical flow between frames $t-1$ and t while $P_{t-1,t}$ denotes warped mask from frames $t-1$ to t. S_0^{ag} and $P_{t-1,t}^{ag}$ denote the instance-agnostic masks defined in Equations (1) and (2). The relative spatial sizes and channel dimensions of feature maps are given. The key feature maps are visualized by summing all channels (best view in color).

attention-guided decoder (Figure 2(c)) to produce segmentation results for the target frame.

3.1. Network Structure

Preliminaries We first present the preliminaries of our overall network structure illustrated in Figure 2. In the semi-supervised video object segmentation setting, the first frame, a.k.a. reference frame I_0, is annotated by human, indicating objects that need to be segmented in the rest of frames. We utilize $I_0 \in R^{H\times W\times 3}$ and $S_0 \in \{0,1\}^{N\times H\times W}$ to represent reference frame and corresponding annotated object segmentation where H and W are image height and width respectively, and N is the total number of object instances annotated in the reference frame. Pixels with values 0 and 1 in S_0 denote background and foreground pixels for each object instance respectively. The target frame t is the frame that needs to be segmented.

Similarly, we utilize $I_t \in R^{H\times W}$, $S_t \in \{0,1\}^{N\times H\times W}$ and $P_t \in [0,1]^{N\times H\times W}$ to represent the target frame t, corresponding ground truth object segmentation mask and object prediction results. Beside, $S_t^P \in \{0,1\}^{N\times H\times W}$ denotes prediction segmentation results in frame t. To equip the system with temporal reasoning capability, we extract optical flow $O_{t-1,t}$ between previous frame I_{t-1} and target frame I_t.

Instance-Agnostic Module We build our instance-agnostic module on top of the architecture proposed in RGMP [26]. To produce the instance-agnostic feature F_t^{ia} for segmenting target frame I_t, the module takes as input the reference frame I_0 with its corresponding agnostic ground-truth mask $S_0^{ag} \in \{0,1\}^{H\times W}$, the target frame I_t with its corresponding agnostic warped mask $P_{t-1,t}^{ag} \in [0,1]^{H\times W}$.

To align previous frame annotation with the target frame, we warp P_{t-1} with flow field $O_{t-1,t}$. The warped mask is denoted as $P_{t-1,t} \in [0,1]^{N\times H\times W}$. For the agnostic mask S_0^{ag} and $P_{t-1,t}^{ag}$, the pixel value of position $(h,w) \in H\times W$ is

$$S_0^{ag}(h,w) = \max_{1\le n\le N} S_0(n,h,w), \tag{1}$$

$$P_{t-1,t}^{ag}(h,w) = \max_{1\le n\le N} P_{t-1,t}(n,h,w). \tag{2}$$

As illustrated in Figure 2(a), the target frame I_t with the agnostic warped mask $P_{t-1,t}^{ag}$ and the reference frame I_0 with the corresponding agnostic mask S_0^{ag} are first processed by a two-stream Siamese encoder, which maps them into two semantic feature maps separately. The two feature maps are then concatenated and decoded to generate instance-agnostic feature maps $F_t^{ia} \in \mathbb{R}^{\frac{H}{8}\times\frac{W}{8}\times 256}$, where H and W are the original image height and width.

In contrast to the original RGMP framework, which processes and segments one object each, we take RGMP as a

Figure 3. Visualization of the instance-specific visual feature and attention feature. For each feature map, the results are obtained by summing the channel dimension and taking the absolute values. For each object instance, the visual feature captures visual texture information. The attention feature highlights the most relevant regions.

generic feature extractor – our method extracts the feature map shared by all instances at one feed-forward path.

Instance-Specific Module To discriminate among different object instances, we propose the instance-specific module, which is a light-weight neural network to encode different object instances into instance visual features $F_t^{vs} \in R^{N\times\frac{H}{4}\times\frac{W}{4}\times 256}$ and instance attention features $F_t^{atn} \in R^{N\times\frac{H}{8}\times\frac{W}{8}\times 256}$. Layer k in F_t^{vs} and F_t^{atn}, denoted as $F_{t,k}^{vs}$ and $F_{t,k}^{atn}$ respectively, represent the corresponding features for the k-th object. To generate instance visual features, the prediction results for the previous frame P_{t-1} is first warped to be aligned with frame t according to the predicted optical flow $O_{t-1,t}$.

The i-th map in the warped object segmentation $P_{(t-1,t),i} \in [0,1]^{H\times W}$ represents the i-th object instance warped mask. Then, the visual-feature extractor takes the channel-wise concatenation of $P_{t-1,t}$ and the target image I_t as input, and produces the instance visual feature $F_t^{vs} \in R^{N\times\frac{H}{4}\times\frac{W}{4}\times 256}$ as illustrated in Figure 2(b). The instance visual feature map is further combined with the down-sampled object instances masks $P_{t-1,t}$, which is subsampled by a ratio of 4 to align with the spatial dimension of F_t^{vs}.

The concatenated features are utilized by the the attention generator (Figure 2(d)) to generate the instance attention feature $F_t^{atn} \in R^{N\times\frac{H}{8}\times\frac{W}{8}\times 256}$. The attention generator aims to generate the instance attention feature in a computational and memory efficient manner, which includes only three convolutional layers as illustrated in Figure 2(d).

The first convolution layer with kernel size 1×1 is utilized to integrate and refine channel-wise input (*i.e.*, F_t^{vs} and down-sampled $P_{(t-1,t)}$). The rest two convolutional layers adopt 3×3 kernel size for aggregating spatial and channel information. The last convolutional layer with kernel size 3×3 and stride 2 is to down-sample the feature map to generate instance attention feature F_t^{atn}, which has the same spatial size as F_t^{ia}. Empirically, we find that removing the activation function after the last convolutional layer produces better results than using ReLU or sigmoid activation function.

Figure 3 presents an example of the instance-specific visual feature $F_{t,k}^{vs}$ and attention feature $F_{t,k}^{atn}$ for each object instance. The visual texture information of each object instance is captured by the visual and attention feature, highlighting the most relevant regions to filter out potentially noisy regions. These two features complement each other to decode the specified object instance from the instance-agnostic feature.

Attention-Guided Decoder Equipped with the instance-specific features, *i.e.*, F_t^{vs} and F_t^{atn}, and instance-agnostic feature F_t^{ia}, we further propose the attention-guided decoder (Figure 2(c)) to separately predict the probability mask $P_{t,k}$ for each object instance k at the target frame t.

First, to introduce instance-specific priors for mining discriminative information from the instance-agnostic feature F_t^{ia} for instance k, we combine F_t^{ia} and instance attention feature $F_{t,k}^{atn}$ to generate extracted instance-agnostic feature $F_{t,k}^{ia}$ for instance k as

$$F_{t,k}^{ia} = F_{t,k}^{atn} \odot F_t^{ia}, \tag{3}$$

where $\odot$ denotes element-wise multiplication, and $F_{t,k}^{atn}$ represents the attention feature for the k-th object. The learned attention generator enables us to obtain the most relevant information for defining the corresponding instance.

Then the extracted instance-agnostic feature $F_{t,k}^{ia}$ is combined with the instance visual feature $F_{t,k}^{vs}$ via a refinement module, which harvests complementary information from instance-agnostic and instance-specific feature maps. The refined feature is further processed by the final prediction module, which has one 3×3 convolution with output of channel dimension 2 and a $4\times$ bilinear up-sampling operation to match the original image resolution. Softmax nonlinearity is finally applied to the output to produce the foreground prediction probability mask $P_{t,k} \in [0,1]^{H\times W}$ for instance k.

Probabilistic Normalization Till now, different object instances are separately predicted. However, they are correlated and constrained by the fact that one pixel can only be assigned to one object instance. To better capture this

intuition, we propose to utilize *softmax aggregation* function [26] to normalize prediction of each pixel considering its object probability among all N object instances.

The probabilities are normalized as

$$\hat{P}_{t,k}(h,w) = \frac{P_{t,k}(h,w)/(1-P_{t,k}(h,w))}{\sum_{i=0}^{N} P_{t,i}(h,w)/(1-P_{t,i}(h,w))}. \quad (4)$$

In this equation, $(h,w) \in \{1,2,...,H\} \times \{1,2,...,W\}$ indicates all pixel locations. $P_{t,0}$ is the background probability map for frame t, which is not predicted in the attention guided module. We derive it by considering all foreground prediction results via

$$P_{t,0}(h,w) = 1 - \max_{1 \le i \le N} P_{t,i}(h,w). \quad (5)$$

The above probabilistic normalization strategy also enables us to directly derive the object segmentation result $S_t^P \in \{0,1\}^{N \times H \times W}$ as Equation (6) without any post-processing:

$$S_t^P(k,h,w) = \mathbf{1}[k = \operatorname*{argmax}_{k \in \{0,1,...,N\}} \hat{P}_{t,k}(h,w)]. \quad (6)$$

$\mathbf{1}[\cdot]$=1 if and only if $\cdot$ is true. The post-processing is adopted in almost all state-of-the-art approaches [15, 7], which needs parameter tuning. In contrast, we simultaneously handle all object instances prediction in our one pass segmentation framework. It enables us to formulate our instance-aware IoU loss function as detailed in Sec. 3.2.

3.2. Training Loss

To train our proposed AGSS-VOS framework, we adopt IoU Loss [12] formulated in Equation (7). $\hat{P}_{t,k}$ and $S_{t,k}$ denote the normalized prediction mask and ground truth mask for instance k in frame t, respectively.

$$\mathcal{L}(\hat{P}_t, S_t) = 1 - \frac{1}{N} \sum_{k=1}^{N} \frac{\sum_{h,w} min(\hat{P}_{t,k}(h,w), S_{t,k}(h,w))}{\sum_{h,w} max(\hat{P}_{t,k}(h,w), S_{t,k}(h,w)))} \quad (7)$$

The IoU loss is utilized to handle large size variation among different object instances since it has similar effect on both small and large objects. Further, it is designed to inspire the network to produce discriminative probability distributions for different instances since it jointly considers probability belonging to all instances.

4. Experiments

We evaluate our approach on challenging Youtube-VOS [28] and DAVIS-2017 [19] datasets. We also perform comprehensive ablation experiments in Section 4.4 to validate the effectiveness of each component, *i.e.*, instance-agnostic module, instance-specific module, and attention-guided decoder.

4.1. Implementation Details

Structure Details The instance-agnostic module is built on top of RGMP [26] except that we take the output of the second refine module as the instance-agnostic feature. The last refinement module is moved to the attention guided decoder. The feature extractor in the instance-specific module consists of two residual blocks.

Training Details In the training phase, we randomly sample a fixed-length sub-sequence in all videos. The first frame in the sampled sequence is utilized as the reference frame. We add two types of data augmentation: 1) flipping each frame horizontally; 2) reversing the sampled sequence. Similar to [26], we use recurrent training scheme to simulate error accumulation and soft mask from the previous frame. Besides, we set a tolerance threshold: if the IoU of previous mask is lower than a threshold, this mask is substituted with the ground truth one since a low-quality mask could misguide the target-frame segmentation. We initialize IAM with pre-trained weights in [26] to accelerate convergence.

Optical flow is calculated with FlowNet-2 [8] whose weights are updated during the training process. The sampled frame is re-sized to 640×320 and the length of the sampled sequence is 8 (frames). We use Adam optimizer and poly learning policy with the initial learning rate $1e-5$ for 10-epoch training. Training on Youtube-Vos training set takes about one day with one NVIDIA TITAN Xp GPU card.

4.2. Evaluation Metrics

The predicted video object segmentation is compared with the ground truth in terms of the following metrics.

- Mask accuracy $\mathcal{J}$: the mean intersection-over-union (mIoU) between the predicted segmentation and ground-truth masks.
- Contour accuracy $\mathcal{F}$: the F-measures of the contour-based precision and recall between the contour points of the predicted segmentation and ground-truth masks.
- Overall score $\mathcal{G}$: the average score of $\mathcal{J}$ and $\mathcal{F}$.

4.3. Comparsion with State-of-the-arts

Youtube-VOS We train our framework on the Youtube-VOS [28] training set, which contains 3,471 videos and approximately half of them contain multiple objects. We evaluate our model on the validation set, which contains 474 videos. The results are evaluated on the open evaluation server [28]. We evaluate on the validation set since the Youtube-VOS test set server is not open.

Youtube-VOS also evaluate $\mathcal{J}, \mathcal{F}$ on seen and unseen objects separately. Objects with categories existing in both

Method	OL	$\mathcal{J}$ seen (%)	$\mathcal{J}$ unseen (%)	$\mathcal{F}$ seen (%)	$\mathcal{F}$ unseen (%)	$\mathcal{G}$ Overall (%)	Time (s)
OSMN [29]		60.0	40.6	60.1	44.0	51.2	0.24
RGMP [26]		59.5	45.2	-	-	53.8	-
S2S [27]		66.7	48.2	65.5	50.3	57.6	0.27
AGAM [9]		66.9	61.2	-	-	66.0	-
MaskTrack [18]	✓	59.9	45.0	59.5	47.9	53.1	20.6
OnAVOS [24]	✓	60.1	46.6	62.7	51.4	55.2	22.3
OSVOS [1]	✓	59.8	54.2	60.5	60.7	58.8	17.2
S2S(+OL) [27]	✓	71.0	55.5	70.0	61.2	64.4	15.4
AGSS-VOS		**71.3**	**65.5**	**75.2**	**73.1**	**71.3**	**0.08**

Table 1. Quantitative results of video object segmentation on Youtube-VOS validation set. 'OL' denotes using online learning. 'time (s)' denotes the running time per frame.

Method	OL	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$	Time (s)
RGMP [26]		64.8	68.6	66.7	0.28
VideoMatch [7]		56.5	-	-	0.35
VideoMatch [7]	✓	61.4	-	-	2.62
OnAVOS [22]	✓	61.0	66.1	63.6	26
PReMVOS [15]	✓	**73.9**	**81.7**	**77.8**	37.4
AGSS-VOS		63.4	69.8	66.6	**0.10**
AGSS-VOS (pre. YTV)		64.9	69.9	67.4	**0.10**

Table 2. Quantitative comparison of different methods on DAVIS-2017 validation set. 'OL' denotes online training. 'pre. YTV' denotes pre-training on Youtube-VOS dataset [28].

Method	OL	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$	Time (s)
RGMP [26]		51.3	54.4	52.8	0.42
OnAVOS [22]	✓	53.4	59.6	56.5	39
PReMVOS [15]	✓	**67.5**	**75.7**	**71.6**	41.3
AGSS-VOS		51.5	57.1	54.3	**0.11**
AGSS-VOS (pre. YTV)		54.8	59.7	57.2	**0.11**

Table 3. Quantitative comparison of different methods on DAVIS-2017 test-dev set. 'OL' denotes online training. 'pre. YTV' denotes pre-training on Youtube-VOS dataset [28].

the training and validation sets are denoted as seen objects, while objects with categories only existing in validation set are denoted as unseen objects.

In Table 1, we show comparison with previous start-of-the-art approaches on Youtube-VOS [28] dataset. Our method achieves a new state-of-the-art of 71.3% in terms of overall scores using only 0.08 second per frame. "OL" in Table 1 denotes online learning in the inference stage. This strategy can help boost the performance. But it is not practical.

Compared with approaches without online learning [29, 27, 9, 26], our AGSS-VOS approach consistently performs better. Moreover, our approach (0.08s/frame) is much faster than the previous efficient approach [29] with 0.24s/frame. Compared with the approaches by online learning [1, 18, 24, 27], our method is $200\times$ times more efficient than compared approaches. In terms of quality, our results are also decent.

DAVIS-2017 DAVIS-2017 [19] contains 60 video sequences for training, 30 sequences for validation and 30 sequences for testing. Most of the video sequences contain multiple objects. The AGSS-VOS model is trained on the DAVIS-2017 training set and evaluated on the validation/test-dev set. Besides, we notice that pre-training on the Youtube-VOS training set and fine-tuning on the DAVIS training set boost the performance.

The comparison with other state-of-the-art methods are demonstrated in Tables 2 and 3. Our method achieves three times faster than the previous fastest approach [26] with comparable accuracy. We note that the accuracy of PReMVOS [13] is higher since it uses online learning.

4.4. Ablation Studies

Analysis of Different Components We do extensive ablation experiments to analyze the effectiveness of different components, *e.g.* instance-agnostic module (IAM), instance-specific module (ISM), probabilistic normalization strategy (NM) and Optical Flow (OF). Quantitative results are illustrated in Table 4.

Table 4 (line 1) shows the result of removing instance-specific module in AGSS-VOS. In this setting, the instance-agnostic feature F_t^{ia} is directly multiplied with warped video object segmentation prediction $P_{t-1,t}$ (Figure 2). The overall score decreases more than 4% compared with AGSS-VOS model (Table 4 (line 5)). As shown in Figure 4, after removing the instance-specific module, the framework fails to enhance the difference between the two horses. It demonstrates usefulness of the proposed instance-specific module for multi-object segmentation in videos.

Table 4 (line 2) shows the quantitative results of our system without instance-agnostic module (IAM). In this setting, the instance-specific features (F_t^{vs} and F_t^{atn} in Figure 2) are directly utilized to produce the object segmentation results. Experimental results show that the overall score drops more than 12% compared with AGSS-VOS model (Table 4 (line 5)). As shown in Figure 4, the segmentation quality becomes much worse without using the instance-

	OF	IAM	ISM	NM	$\mathcal{J}$ seen (%)	$\mathcal{J}$ unseen (%)	$\mathcal{F}$ seen (%)	$\mathcal{F}$ unseen (%)	$\mathcal{G}$ overall (%)
1	✓	✓		✓	69.5	59.3	73.2	66.1	67.0
2	✓		✓	✓	60.2	51.7	63.0	59.7	58.6
3	✓	✓	✓		69.3	61.3	73.7	70.0	68.6
4		✓	✓	✓	69.9	59.9	73.9	67.2	67.8
5	✓	✓	✓	✓	**71.3**	**65.5**	**75.2**	**73.1**	**71.3**

Table 4. Ablation study of component effect on the Youtube-VOS [28] dataset. 'OF' denotes warping the previous mask to the target frame via optical flow. 'IAM' denotes the instance-agnostic module. 'ISM' denotes the instance-specific module. 'NM' denotes probabilistic normalization.

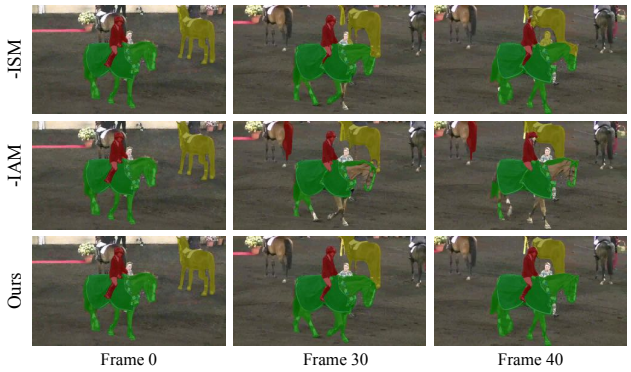

Figure 4. Effect of the Instance-Agnostic Module (IAM) and the Instance-Specific Moodule (ISM). '-ISM' and '-IAM' denote removing the instance-specific module and instance-agnostic module respectively.

Figure 5. Effect of probabilistic normalization (NM). '-NM' denotes the abandonment of probabilistic normalization. Without normalization, segmentation of the backpack strap cannot be retained.

agnostic module. It demonstrates that the instance-agnostic module actually learns crucial information for video object segmentation.

Table 4 (line 3) shows the result by removing the probabilistic normalization process and directly utilizing the output for training. The performance drops by 2% compared with Table 4 (line 5). Figure 5 demonstrates the effect of probabilistic normalization. By normalizing the probability of each prediction, the AGSS-VOS model is able to retain segmentation of small objects *e.g.* the backpack strap, in a long range of frames.

In addition, we evaluate utilizing optical flow to align the

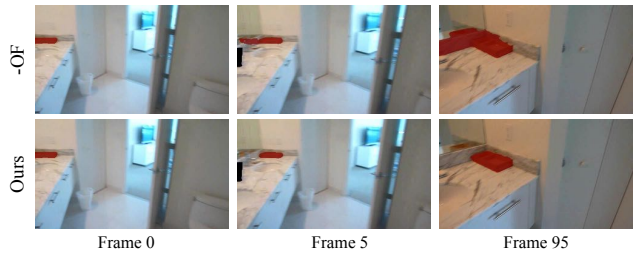

Figure 6. Effect of optical flow (OF). '-OF' denotes the abandonment of optical flow. Without optical flow, the box in the mirror is segmented mistakenly.

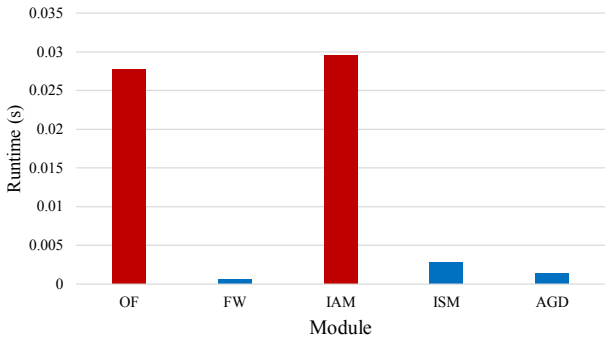

Figure 7. Illustration of running time of each module in AGSS-VOS in case of one-object with one-feed-forward. OF and FW denote calculation of optical flow and warping the previous mask using optical flow. IAM and ISM denote the instance-agnostic module and instance-specific module, respectively. AGD denotes attention-guided decoder.

previous segmentation prediction P_{t-1} to the current frame $P_{t-1,t}$. Experimental results without optical flow alignment drop by 3% as shown in Table 4 (line 4). Figure 6 demonstrates the effect of optical flow. Without aligning the previous frame's mask, the AGSS-VOS model segments the box in the mirror mistakenly. This demonstrates that utilizing optical flow to align the input helps the system better segment objects in motion scenarios.

Runtime Analysis We show the running time of each module of AGSS-VOS in the case of one-object with one-

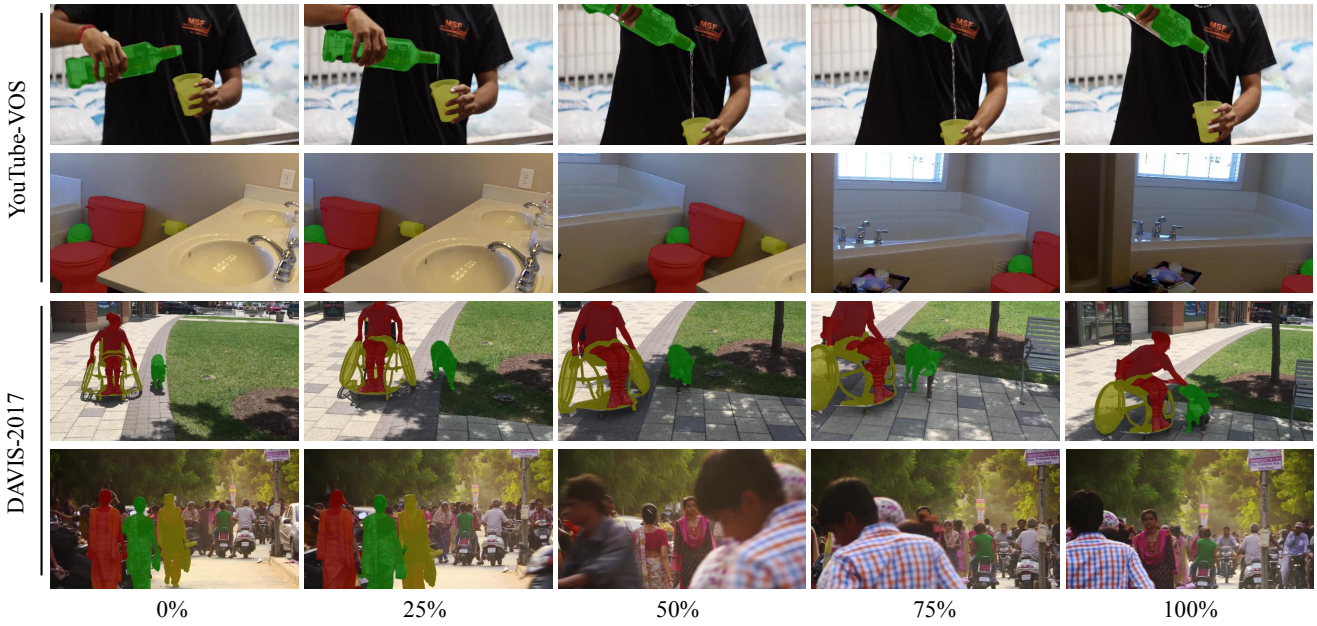

Figure 8. Illustration of the result of our method on the DAVIS-2017 and YouTube-VOS datasets. Frames are sampled uniformly. The last row shows failure mode of our approach.

feed-forward in Figure 7. Optical flow computation (OF) and instance-agnostic module (IAM) occupy more than 92% of the total computational time. This part of computation time does not increase along with the number of object instances since operations only need to be computed once for one frame regardless of the number of object instances.

While the optical flow warping (FW), instance-specific module (ISM), and attention guided decoder (AGD) need to be computed for each instance separately, they only occupy less than 8% of the computation time. Benefited from rich representation of the instance-agnostic module, we design the light-weight instance-specific module capturing rough position information of the instances represented as attention maps. The whole system gains high efficiency in processing multiple objects in one path without sacrificing accuracy.

4.5. Qualitative Results

Qualitative results on DAVIS-2017 [19] and Youtube-VOS [28] datasets are shown in Figure 8. These sequences all contain multiple objects with diverse motion, shape, and size. Our AGSS-VOS produces high-quality results in these challenging scenarios. For example, our system can successfully segment the small moving bottle in Figure 8 (the first row) – noticing part of it moves out of the screen in some frames. In the last row, AGSS-VOS fails to segment people after occlusion. The challenging scenarios can be addressed by incorporating multiple guidance frames, or re-identification techniques [15, 11], which will be our future direction.

5. Conclusion

In this paper, we have proposed AGSS-VOS for single-shot video-object segmentation. Our framework includes an instance-agnostic module, an instance-specific module and an attention-guided decoder. The instance-agnostic module extracts the instance-agnostic feature for all the objects, while the instance-specific module generates the instance-specific visual and attention features for each object, represented as attention maps. In the attention-guided decoder, the instance-agnostic feature is multiplied by the instance attention features, which are further refined with the instance visual feature to produce prediction of each object. Moreover, we have designed the probabilistic normalization strategy to enable end-to-end optimizing scores of all instances. Our system is reasonably accurate and quite efficient compared with previous state-of-the-art methods especially when multiple objects exist in the videos.

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*, 2017.

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

[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 *CVPR*, 2018.

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

[5] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017.

[6] Yuan-Ting Hu, Jia-Bin Huang, and Alexander Schwing. Maskrnn: Instance level video object segmentation. In *NeurIPS*, 2017.

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

[8] 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 *CVPR*, 2017.

[9] Emil Brissman Fahad Shahbaz Khan Joakim Johnander, Martin Danelljan and Michael Felsberg. A generative appearance model for end-to-end video object segmentation. *arXiv:1811.11611*, 2018.

[10] Anna Khoreva, Rodrigo Benenson, Eddy Ilg, Thomas Brox, and Bernt Schiele. Lucid data dreaming for object tracking. In *The DAVIS Challenge on Video Object Segmentation*, 2017.

[11] Xiaoxiao Li and Chen Change Loy. Video object segmentation with joint re-identification and attention-aware mask propagation. *arXiv:1803.04242*, 2018.

[12] Zhuwen Li, Qifeng Chen, and Vladlen Koltun. Interactive image segmentation with latent diversity. In *CVPR*, 2018.

[13] Jonathon Luiten, Paul Voigtlaender, and Bastian Leibe. Premvos: Proposal-generation, refinement and merging for the davis challenge on video object segmentation 2018. In *The 2018 DAVIS Challenge on Video Object Segmentation-CVPR Workshops*, 2018.

[14] Jonathon Luiten, Paul Voigtlaender, and Bastian Leibe. Premvos: Proposal-generation, refinement and merging for the youtube-vos challenge on video object segmentation 2018. In *The 1st Large-scale Video Object Segmentation Challenge-ECCV Workshops*, 2018.

[15] Jonathon Luiten, Paul Voigtlaender, and Bastian Leibe. Premvos: Proposal-generation, refinement and merging for video object segmentation. *arXiv:1807.09190*, 2018.

[16] K.-K. Maninis, S. Caelles, Y. Chen, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. Van Gool. Video object segmentation without temporal information. *TPAMI*, 2018.

[17] Voigtlaender Paul, Chai Yuning, Schroff Florian, Adam Hartwig, Leibe Bastian, and Chen Liang-Chieh. Feelvos: Fast end-to-end embedding learning for video object segmentation. 2019.

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

[19] 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.

[20] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015.

[21] Pavel Tokmakov, Karteek Alahari, and Cordelia Schmid. Learning video object segmentation with visual memory. In *ICCV*, 2017.

[22] Paul Voigtlaender, Michael Krause, Aljosa Osep, Jonathon Luiten, Berin Balachandar Gnana Sekar, Andreas Geiger, and Bastian Leibe. Mots: Multi-object tracking and segmentation. *arXiv:1902.03604*, 2019.

[23] Paul Voigtlaender and Bastian Leibe. Online adaptation of convolutional neural networks for the 2017 davis challenge on video object segmentation. In *The 2017 DAVIS Challenge on Video Object Segmentation-CVPR Workshops*, 2017.

[24] Paul Voigtlaender and Bastian Leibe. Online adaptation of convolutional neural networks for video object segmentation. *arXiv:1706.09364*, 2017.

[25] Qiang Wang, Li Zhang, Luca Bertinetto, Weiming Hu, and Philip HS Torr. Fast online object tracking and segmentation: A unifying approach. *arXiv:1812.05050*, 2018.

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

[27] 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*, 2018.

[28] Ning Xu, Linjie Yang, Yuchen Fan, Dingcheng Yue, Yuchen Liang, Jianchao Yang, and Thomas Huang. Youtube-vos: A large-scale video object segmentation benchmark. *arXiv:1809.03327*, 2018.

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