Pyramid Object Association with Transformers for Video Object Segmentation

Yuanyou Xu
yuanyouxu515@gmail.com

Abstract

For semi-supervised video object segmentation task, Associating Objects with Transformers (AOT) [21, 23] has been proven to be outstanding under multiple object segmentation scenarios. Based on AOT, this paper investigates the more powerful architecture and the more efficient transformer block. A novel pyramid architecture (PAOT) is designed to fully utilize the multi-scale features. With the architecture, higher performance is achieved by deeper transformers associating objects on multiple scales. While the architecture brings better performance, the computing burden increases vastly. In order to control the computational cost and realize better efficiency, an efficient version of the long short-term transformer (ELSTT) is proposed. The head reduction strategy and dilated attention mechanism in the ELSTT not only reduce the computational time but also cut down the demand for the memory space. Therefore, the pyramid architecture and the efficient LSTT enable PAOT to be superior both in performance and efficiency. After applying test-time augmentations and model ensemble, we rank 2nd in Track 1 (Video Object Segmentation) of the 4th Large-Scale Video Object Segmentation Challenge.

1. Introduction

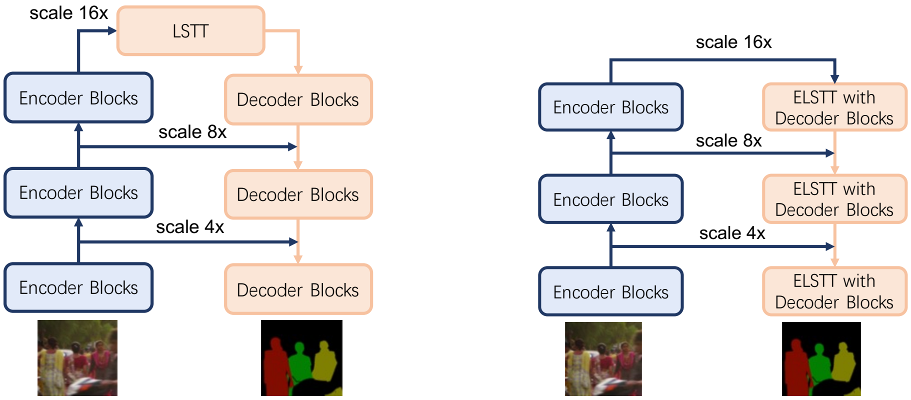
Video object segmentation (VOS) is a fundamental task in computer vision. There are several different settings for video object segmentation. In this paper, we mainly focus on semi-supervised video object segmentation. In this task, target objects are specified by one or more reference frames with pixel-level masks in a video. Semi-supervised video object segmentation aims to segment all target objects in all frames in the video.

Semi-supervised video object segmentation has been deeply explored in recent few years, especially via learning-based techniques. A straightforward idea is to match pixels between frames to acquire information about target objects. FEELVOS [17] uses the global and local matching between pixel-wise embeddings to transfer information through frames. CFBI(+) [20,22] considers the background matching equally as the foreground matching, and thus a foreground-background integration approach is proposed.

To tackle the problems in VOS, the similarity of the objects in both the spatial and temporal spaces should be fully utilized. As a milestone, STM [13] introduces the memory networks to video object segmentation and models the matching as space-time memory reading. Based on the space-time memory reading framework, some of the later works design better memory reading or matching methods. KMN [14] proposes the kernelized memory network, which adds the kernel constraint on the memory reading to meet the local assumption. LCM [7] learns position consistency in global memory matching and introduces target consistency in local memory matching, which makes the segmentation more robust and reliable.

AOT [21, 23] introduces the transformer structure to VOS and develops a quite different framework from STM. Unlike previous methods which segment multiple objects one by one and merge results by post ensemble, AOT processes all the target objects simultaneously with its multi-object identification mechanism. To model multi-object association, a long short-term transformer (LSTT) is designed for constructing memory matching and mask propagation. AOTv2 [19] improves the identification-based attention in the LSTT block by coupling identification and vision embeddings in different embedding spaces for different layers.

The fact that the scale of an object in a video usually changes over time is a common challenge in VOS. Despite the effective designs in the AOT model, only the feature maps on the smallest scale are fed into the LSTT module, which follows the common way to obtain the frame embeddings in other frameworks. We argue that using single-scale feature maps for matching in some cases is not sufficient enough to achieve robust and reliable results. In addition, the architecture of AOT can result in performance saturation. Increasing the number of LSTT layers from one to two or three in the AOT gains performance boost. However, further increase in LSTT layers contributes little to the performance. To tackle these problems, we propose a novel pyramid architecture to associate objects with transformers (PAOT), which sequentially merges feature maps on multiple scales to form the feature pyramid and finishes matching and propagation in a progressive manner. Com-



(a) The architecture of AOT.

(b) The architecture of PAOT.

Figure 1. The above two pictures compare the architectures of AOT and PAOT. The main difference is that the LSTT module is combined with the decoder to form a pyramid architecture in the PAOT.

pared with AOT, PAOT organizes richer encoded features for matching, and also breaks through the limit of AOT as it enables deeper transformer layers. Therefore, PAOT achieves higher performance and better robustness.

Although the architecture brings higher performance, the computational cost increases vastly since the large scale feature maps are involved in. To reduce the computational cost, we propose the efficient long short-term transformer (ELSTT). In the ELSTT, the number of heads in the long-term attention is reduced. In addition, the dilated attention mechanism is employed to ease the huge memory consumption caused by the attention between large size keys and values. Finally, the pyramid architecture is combined with the ELSTT to complete the PAOT model, which has both high performance and efficiency.

2. Method

In this section, the main method we use is elaborated on. First, we shortly revisit the AOT model to prepare the reader for the following content. Next, the architecture design of our pyramid AOT model is introduced. Last but not least, we present the details of the efficient long short-term transformer in PAOT.

2.1. Revisit AOT

To make the model capable of handling multiple objects at the same time, an identification mechanism is proposed in the AOT. First, identification embedding is used to embed the masks of multiple different targets into the same feature space for propagation. Assuming N targets are in the video scenery, an identity bank which contains $M(M > N)$ identification vectors is used to assign identities to different objects randomly. After the identity assignment, each different target has a different identification embedding, and thus the model can propagate all the target identification information from memory frames to the current frame by attaching the identification embedding to the visual features.

The Long short-term transformer (LSTT) is one of the core modules in the AOT. Following the common transformer blocks [2, 16], the LSTT firstly employs a self-attention layer, which is responsible for learning the association or correlation among the targets within the current frame. Then, the LSTT additionally introduces a long-term attention for aggregating targets' information from long-term memory frames and a short-term attention for learning temporal smoothness from nearby short-term frames. The final module is a common 2-layer feed-forward MLP with GELU [6] non-linearity in between. The multi-object information will be gradually aggregated and associated as the LSTT structure goes deeper, leading to more accurate attention-based matching. More analysis can be found in [21].

2.2. Pyramid AOT

2.2.1 Architecture

The whole architecture of PAOT as shown in Figure 1b follows the encoder-decoder design like many classical segmentation networks, U-Net for example. In the encoder part, it is divided into several blocks. After each block, the feature maps of the input are down-sampled to smaller sizes. As a result, the outputs of these encoder blocks can provide features on different scales.

In the decoder part, unlike the FPN module [8] in the AOT as shown in Figure 1a, the LSTT is combined with several decoder blocks to form the pyramid stages of PAOT. The feature maps on each scale from the encoder are fed into the corresponding stage. The ELSTT module in the stage is responsible for performing matching between current frame and memory frames and aggregating mask information from memory frames. Next, the decoder blocks are able to decode the information. Such a stage repeats for several times on different scales. Such a design has two advantages. First, it fully utilizes the feature maps on different scales. The FPN module in the AOT also takes feature maps on different scales as its inputs. However, they just function as shortcut connections to form the residual structure. Only the feature maps on the smallest scale are fed into the LSTT module and perform matching across memory frames. The pyramid architecture enables feature maps on multiple scales to be involved in matching. Second, it deepens the whole model and the model capacity also extends along with the depth. In the AOT, increasing the number of LSTT layers from one to two or three gains performance boosts. However, further increase in LSTT layers contributes little to the performance. While in PAOT, each stage includes several LSTT blocks and the information aggregated by the previous stage can be accumulated and reused in the current stage. As a result, the number of LSTT layers increases from three to five and the model continues to gain performance improvement.

2.2.2 Efficient Long Short-Term Transformer

The basic structure of the LSTT is introduced in 2.1. Directly using this design in the pyramid structure causes two problems. First, the increase in the number of transformer layers leads to the increase in the amount of calculation. Second, the use of large scale feature maps in attention causes great demand for the memory space. Taking these problems into account, we design a more efficient structure for the long short-term transformer (ELSTT).

After thorough analysis of the computational cost of each part in the LSTT, we find the long-term attention dominates. The long-term attention calculates the correlation between current frame and memory frames. It needs longer time and larger memory space to finish the matrix multiplication between the query of current frame and the key of memory frames as the number of memory frames increases. In order to cut down the computational cost, we reduce the head number in the long-term attention. The long-term attention in the AOT is a multi-head attention with 8 heads. We reduce the number of heads from 8 to 4 or even 1. Owing to the head reduction, the whole model speeds up twice and thus more LSTT layers can be afforded in the same running time.

However, the attention module with fewer heads still needs a large amount of memory to compute if the sizes of the key and the value are large. In the ELSTT, we apply dilated attention to save the memory space. In detail, we dilate the key and the value before computing the self attention and long-term attention for large scales:

$$K_s = s(K), V_s = s(V)$$

$$h = Softmax(\frac{QK_s^T}{d_h})V_s.$$

K_s is the down-sampled version of the original key, and V_s is the down-sampled version of the original value. $s(\cdot)$ is the down-sampling function. The actual attention performs among Q, K_s and V_s instead of Q, K and V. After down-sampling, the width and height of the key and the value both become half. The dilated attention not only saves lots of memory space but also accelerates calculation.

3. Experiments

In this section, we introduce the settings of the experiments and exhibit our results.

3.1. Training

In PAOT, the backbones for the encoder are chosen in ResNet-50 [5] and Swin Transformer-Base [10]. While ResNet-50 is more lightweight than Swin Transformer-Base, the later achieves higher performance. As for the decoder, the PAOT model includes ELSTT modules in multiple stages. In practice, we choose the number of the layers in the ELSTT in three stages $16\times, 16\times, 8\times$ to be 3,1,1 respectively. Note that we do not use the $4\times$ scale feature maps in terms of the computational resource, and instead we duplicate the $16\times$ scale feature maps twice to form the feature pyramid.

The training stage is divided into two phases: (1) pre-training on the synthetic video sequences generated by static image datasets [1, 3, 4, 9, 15] by randomly applying multiple image augmentations [12]. (2) main training on the real video sequences by randomly applying video augmentations [20].

The training set includes two parts. The first is YouTube-VOS [18] training set. It contains 3471 videos with 65 categories. To reinforce the capability of generalization of the model, the VIPSeg dataset [11] is also incorporated into the training set. VIPSeg contains 3536 videos with 58 thing classes and their frames are annotated in a panoptic manner. We convert the annotations into the suitable format for video object segmentation for training.

During the PAOT training, we use 4 Tesla A100 GPUs, and the batch size is 16. For pre-training, we use an initial learning rate of 4×10^{-4} for 100,000 steps. For main training, the initial learning rate is set to 2×10^{-4}, and the training steps are 100,000. The learning rate gradually decays to 1×10^{-5} in a polynomial manner [20].

3.2. Evaluation

We evaluate our model on YouTube-VOS [18] validation set which contains 474/507 videos in the 2018/2019 version with additional 26 unseen categories. The unseen categories do not exist in the training set in order to evaluate the generalization ability of algorithms.

Method	$J\&F$	J_seen	F_seen	J_unseen	F_unseen
SwinB-AOTv2L [19]	85.2	84.2	88.9	79.8	88.0
SwinB-PAOT	85.9	85.3	89.9	80.4	88.0
+VIPSeg [11]	86.2	84.7	89.5	81.2	89.2
+Full Frames	87.1	85.7	90.5	82.2	90.1
+MS_Flip	87.4	86.1	90.9	82.5	90.3
+Ensemble	87.9	86.6	91.5	82.8	90.6

Table 1. Ablation study on YouTube-VOS 2019 validation set. The PAOT model uses Swin Transformer-Base as the backbone and 4-head ELSTT.

When evaluating, all the videos are restricted to be not bigger than $1.3 \times 480p$ resolution [20–22]. As for test-time augmentations, both multi-scale test and flip test are used. The scales are $\{1.2\times, 1.3\times, 1.4\times\}$ and each scale includes non-flipped and flipped test. If the full frame version of the videos are provided, the model can run on 5 FPS videos instead of 1 FPS ones for better performance. Since denser frames offer more detailed spatial motion clues, it is easier for the model to propagate the masks.

The evaluation metric is the J score, calculated as the average Intersect over Union (IoU) score between the prediction and the ground truth mask, and the F score, calculated as an average boundary similarity measure between the boundary of the prediction and the ground truth, and their mean value, denoted as $J\&F$ as the overall metric. We evaluate all the results on official evaluation servers.

3.3. Results

In the Track 1 (Video Object Segmentation) of 4th Large-Scale Video Object Segmentation Competition, we rank 2nd place on the test set. The leaderboard is shown in Table 2.

We conduct experiments in the way of ablation study to demonstrate the effectiveness of the methods we use. The results are shown in Table 1. We set AOTv2L [19] as the baseline and the backbone is Swin Transformer-Base [10]. It is trained on the YouTube-VOS training set only. When the architecture is replaced with PAOT, the performance increases from 85.2 to 85.9. With VIPSeg added to the training set, the overall performance reaches 86.2. Evaluating with full frames boosts the performance to 87.1. The multi-scale and flipped test-time augmentations bring 0.3 increase. Five models are used for the ensemble to obtain the final result. All models are PAOT and one of them has a ResNet-50 backbone with 8-head ELSTT and all other 4 models have Swin Transformer-Base backbones. For the 4 models with Swin Transformer-Base backbone, two of them have 1-head ELSTT and the other two have 4-head ELSTT. The difference between the two 1-head models is that one loads ImageNet classification pre-trained backbone and the other loads ImageNet object detection pre-trained backbone for the pre-training stage, as well as the two 4-head models.

Team Name	$J\&F$	J_seen	J_unseen	F_seen	F_unseen	Ranking
Thursday_Group	0.872 (1)	0.855 (1)	0.817 (3)	0.914 (1)	0.903 (1)	1
ux (ours)	**0.867 (2)**	**0.844 (3)**	**0.819 (1)**	**0.903 (2)**	**0.903 (2)**	**2**
zjmagicworld	0.862 (3)	0.841 (4)	0.816 (4)	0.895 (4)	0.896 (4)	3
whc	0.862 (4)	0.840 (5)	0.818 (2)	0.894 (5)	0.896 (5)	4
gogo	0.861 (5)	0.847 (2)	0.808 (7)	0.901 (3)	0.890 (6)	5
sz	0.857 (6)	0.831 (6)	0.815 (5)	0.886 (7)	0.896 (3)	6

Table 2. The leaderboard for the Track 1 (Video Object Segmentation) of 4th Large-Scale Video Object Segmentation Competition. We rank 2nd place in the competition.

References

[1] Ming-Ming Cheng, Niloy J Mitra, Xiaolei Huang, Philip HS Torr, and Shi-Min Hu. Global contrast based salient region detection. *IEEE transactions on pattern analysis and machine intelligence*, 37(3):569–582, 2014. 3

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2

[3] 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. 3

[4] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *2011 international conference on computer vision*, pages 991–998. IEEE, 2011. 3

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3

[6] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 2

[7] Li Hu, Peng Zhang, Bang Zhang, Pan Pan, Yinghui Xu, and Rong Jin. Learning position and target consistency for memory-based video object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4144–4154, 2021. 1

[8] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 2

[9] 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 *European conference on computer vision*, pages 740–755. Springer, 2014. 3

[10] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 3, 4

[11] Jiaxu Miao, Xiaohan Wang, Yu Wu, Wei Li, Xu Zhang, Yunchao Wei, and Yi Yang. Large-scale video panoptic segmentation in the wild: A benchmark. In *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 21033–21043, 2022. 3, 4

[12] Seoung Wug Oh, Joon-Young Lee, Kalyan Sunkavalli, and Seon Joo Kim. Fast video object segmentation by reference-guided mask propagation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7376–7385, 2018. 3

[13] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9226–9235, 2019. 1

[14] Hongje Seong, Junhyuk Hyun, and Euntai Kim. Kernelized memory network for video object segmentation. In *European Conference on Computer Vision*, pages 629–645. Springer, 2020. 1

[15] Jianping Shi, Qiong Yan, Li Xu, and Jiaya Jia. Hierarchical image saliency detection on extended cssd. *IEEE transactions on pattern analysis and machine intelligence*, 38(4):717–729, 2015. 3

[16] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2

[17] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9481–9490, 2019. 1

[18] 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 preprint arXiv:1809.03327*, 2018. 3

[19] Zongxin Yang, Jiaxu Miao, Xiaohan Wang, Yunchao Wei, and Yi Yang. Associating objects with scalable transformers for video object segmentation. *arXiv preprint arXiv:2203.11442*, 2022. 1, 4

[20] Zongxin Yang, Yunchao Wei, and Yi Yang. Collaborative video object segmentation by foreground-background integration. In *European Conference on Computer Vision*, pages 332–348. Springer, 2020. 1, 3, 4

[21] Zongxin Yang, Yunchao Wei, and Yi Yang. Associating objects with transformers for video object segmentation. *Advances in Neural Information Processing Systems*, 34, 2021. 1, 2, 4

[22] Zongxin Yang, Yunchao Wei, and Yi Yang. Collaborative video object segmentation by multi-scale foreground-background integration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 1, 4

[23] Zongxin Yang, Jian Zhang, Wenhao Wang, Wenhua Han, Yue Yu, Yingying Li, Jian Wang, Yunchao Wei, Yifan Sun, and Yi Yang. Towards multi-object association from foreground-background integration. In *CVPR Workshops*, volume 2, 2021. 1