Feature Aligned Memory Network for Video Object Segmentation

Wangwang Yang[1], Zhengyi Lv[2], Jiangyu Liu[2], and Di Huang[1]

[1]Beihang University
[2]Megvii Research

Abstract

In this paper, we propose a Feature Aligned Memory Network(FAMNet) for semi-supervised video object segmentation. Unlike other variants of Space-Time Memory Networks where two different encoders are used for query frame and memory frame separately, our method only uses one encoder for both query frame and memory frame, regarding VOS as a pixel-level matching task completely. By sharing the encoder weights, we not only reduce the amount of model parameters, but also make the extracted features more aligned, which is conducive to subsequent pixel-level matching. Besides, We designed a crop then zoom strategy to improve the accuracy of small objects and relieve the impact of accumulated errors caused by previous frames. Through additional multi-scale testing and model ensemble, our FAMNet achieves the fifth place on the YouTube-VOS 2021 Semi-supervised Video Object Segmentation Challenge with a J&F mean score of 83.9%.

1. Introduction

Video Object Segmentation (VOS) is a fundamental task in computer vision and has attracted more and more attention in the community. It can perform accurate pixel-level segmentation for the objects in the video so that it has a wide range of applications in video editing, video understanding, automatic driving, etc. In this paper, we focus on semi-supervised video object segmentation(Semi-VOS), which targets at segmenting particular object instances throughout the entire video sequence given only the object masks of the first frame.

YouTube-VOS[17] is a popular video object segmentation benchmark. Challenges like fast motion, occlusion, large appearance variation, multiple similar objects disturbing each other etc. appear in YouTube-VOS dataset frequently. Compared with another popular benchmark DAVIS[12], YouTube-VOS has some scenarios where only part of the object appear in the first frame bringing additional difficulties and ambiguities for segmenting the object in subsequent frames.

In recent years, most of the advanced Semi-VOS models can be regarded as matching-based methods. Some of them[7][15] only use the first frame as the template and predict objects at each frame independently since the object's mask in the first frame come from the given reliable groundtruth. But these approaches have difficulties in dealing with large appearance variation and complex motion which appear in the YouTube-VOS dataset frequently. There are also some methods[11][6] only using the previous frame as the reference because of the great similarity between the previous frame and the current frame, but they suffer from occlusions and object out of view unfortunately. [14][18] use both the first frame and the previous frame for global matching and local matching separately, combining the advantages of the above two types of methods. In order to use more reference frames, Space-Time Memory network(STM)[10] establishes a memory bank and memory read mechanism so that the current query frame can be matched with the first frame, the previous frame and other intermediate frames at the pixel level. We develop our method based on STM because of its simplicity and effectiveness.

In STM, there are some restrictions that affect the further improvement of the performance, so we made several improvements for better result. Firstly, STM encodes query frame and memory frame with two disparate encoders separately, because memory frame's input include RGB image and corresponding mask while query frame only has RGB image as input. In fact, the two encoders have basically the same network structure(such as ResNet50 network) except for the number of input channels. The implement of this two separate encoders not only brings redundant model parameters which is inefficient, but also makes the features of query frame and memory frame inconsistent that adding extra difficulties to the subsequent pixel-level matching stage. To solve this problem, we make a simple modification to the

input of the two encoders, so that they can share the weights and produce more aligned features for better memory read operation. Secondly, STM uses feature map with a resolution 1/16 of original input size for pixel level feature matching, which is very unfriendly to small objects. In addition, STM only considers the pixel level feature matching without considering the object level information, thus losing the object-centric information. Therefore, confusion will occur while facing the scene of multiple similar objects, and sometimes there will also be some extra error segmentation caused by accumulated errors in the background area. We introduce an additional tracker as an aid and design a crop then zoom strategy to alleviate the above problems. Thirdly, original STM neglect the spatial relationship between pixels in the query frame so we employ an Atrous Spatial Pyramid Pooling (ASPP) module[2] before the network's decoder to capture more multi-scale context information.

2. Method

2.1. Aligning Features through Weight Shared Encoder

Our method uses a weight shared encoder for both query frame and memory frame like a siamese network structure. In order to solve the problem of mismatch between the two encoder inputs, we make a simple modification to the input of the encoder. For memory frame, we concatenate the RGB image with its corresponding foreground object's mask to form a 4-channel input. Here we are different from original STM's memory encoder which has a 5-channel input(3 channel for RGB image, 1 channel for current foreground object's mask and another 1 channel for the union of other foreground objects' masks). For query frame, we concatenate the RGB image with an all-one mask so the input of query frame and memory frame become exactly the same in form and they can share the single encoder to extract the feature for pixel level matching. Through padding the all-one mask for query image, the pixels in the foreground area are easier to be matched. Besides, the feature space mapped to the memory frame and query frame is exactly the same, since only one encoder is used, which reduces the difficulty of model training. In our experimental observations, after making such modifications to the encoder input, our method can memory and match the foreground and background simultaneously which conducive to segmenting the unseen texture area of the object due to large deformation or moving from border. We show some typical examples in Figure 1.

2.2. Crop then Zoom Strategy

Semi-VOS and visual object tracking(VOT) are conceptually similar. When the rectangle box in the VOT is replaced with a pixel-level mask, the two tasks are exactly the same in form. The main difference between Semi-VOS and VOT is the application scenario. Semi-VOS task usually consider larger objects, and attach great importance to the non-rigid deformation of the objects, while VOT task usually consider smaller objects whose motion range is larger and the scene is more challenging. Since VOT uses a rectangle box to represent the tracked object, it naturally has object-level information which is what STM network lacks. Therefore, we combined the output of a tracker(we use an advanced transformer-based tracker[3] proposed recently) and the output of the VOS model to design a crop then zoom strategy to further improve the performance of the model, especially in scenes with small objects and accumulated errors. The pipeline is depicted in Figure 2. Specifically, we first use the tracker to track the object, and then crop the area where the object is located, and then resize the patch to a fixed larger resolution. After that we send it to the original VOS pipeline. Finally, the segmentation result will be remapped back to the original image. Note that we did not use the crop then zoom strategy for all objects in the dataset. We apply this strategy to objects whose size is always smaller than 200x200. Since the result of the tracker is not always completely correct, and it is necessary to judge whether the object is always smaller than the preset size, we first use the VOS model to perform preliminary segmentation of the video objects, and then combine the output result of the tracker to select the suitable objects to apply the crop then zoom strategy. Figure 3. includes some examples to prove the effectiveness of this strategy.

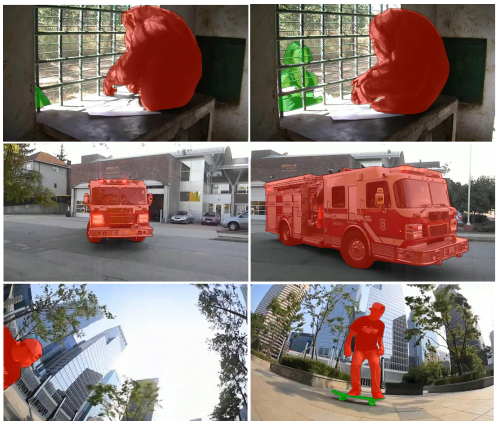

Figure 1. Some typical examples produced by our weight shared encoder model. The left column is the first frames with given masks and the right column shows our robust segmentation results even though the object changes significantly or only partially appears in the initial frame.

2.3. More Local Context Information

It is impossible to obtain accurate segmentation results only by matching pixel-level features. The original STM

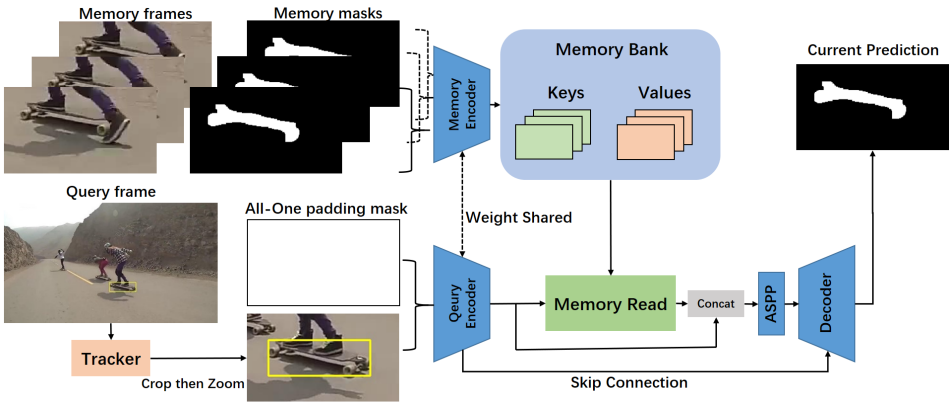

Figure 2. Overview of the pipeline of Feature Aligned Memory Network.

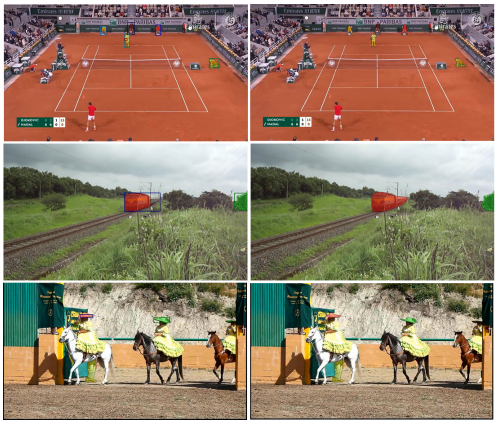

Figure 3. Several visualization comparisons about our crop then zoom strategy. The left column is the initial segmentation results with the tracker's output. The right column shows our fine-grained segmentation results. Best viewed with zoom-in.

network uses a decoder with skip connections to obtain final segmentation results. In order to further increase the receptive field and capture multi-scale local context information, we add an ASPP module[2] before the decoder.

3. Experiments

3.1. Training Details

We follow the training settings in STM[10] which using a two-stage training strategy. We use several saliency and semantic segmentation datasets including MSRA10K[4], ECSSD[13], PASCAL VOC[5], COCO[9] as pre-training data, and YouTube-VOS 2019 training set as main-training data.

During pre-training, we use static images to synthesis virtual video sequence because of the lack of training data. In our experiments, we did find that pre-training is very useful. 384x384 patches are randomly cropped from static images and data augmentation like random scale, random affine transform, random horizontal flip, color jitter are applied to simulate the change of objects in the video. During main-training, we crop 384x640 patches from YouTube-VOS dataset following[19]. In order to reduce low-quality samples caused by random crop, we only apply random crop near the foreground area. For each video, we randomly sample three temporally-ordered frames with a maximum interval 25 and reverse the sequence with a probability of 50%.

We set the minibatch size to 6 per GPU for pre-training and 3 per GPU for main-training and disabled all the batch normalization layers in backbone as[10]. We minimize the cross-entropy loss and Lovász-Softmax loss[1] using Adam optimizer[8] with a initial learning rate of 1e-5. StepLR learning rate schedule is applied to make the model converge stably. All training are finished using four NVIDIA GeForce RTX 2080Ti GPUs.

To alleviate the shortcomings of the Adam optimizer that is easy to converge to the sub-optimal solution, we switch the Adam optimizer to SGD following[16] and fine-tune the model for another 50 epochs with a learning rate of 1e-4 after the main-training.

3.2. Testing Details

We adopt flip and multi-scale testing to segment objects that vary in scales. We also use model ensemble to further improve the performance of our model. Specifically, we use two backbone networks ResNet50 and ResNeXt50 whose performances are similar in our pipeline. For each model, we use two different memory frame sampling strategies for predicting because some objects appear in very few frames in YouTube-VOS. The probabilities of the four prediction were simply averaged as the final result. As shown in Table 2, our method achieves the fifth place on the Challenge.

3.3. Ablation Study

We study the contribution of all the components and tricks in our method. The quantitative results are shown in Table 1. We boost the performance of STM network to 85.3% on YouTube-VOS 2019 validation set finally.

Components	Overall
Baseline(Re-implementation STM)	80.6
+ ASPP Module	81.1
+ Weight Shared Encoder	83.0
+ Multi-scale & Flip Testing	83.7
+ Switch Adam to SGD Training	84.0
+ Crop then Zoom Strategy	84.7
+ Model Ensemble	85.3

Table 1. Ablation study on YouTube-VOS 2019 validation set.

4. Conclusion

In this paper, we propose a Feature Aligned Memory Network(FAMNet) that improved some problems in original STM network, making the performance of the STM network reach a new level. In the end, our method achieves the fifth place on the YouTube-VOS 2021 Semi-supervised Video Object Segmentation Challenge with an overall score of 83.9%.

Team Name	Overall	J_{seen}	J_{unseen}	F_{seen}	F_{unseen}
wenhaowang	0.856(1)	0.836(2)	0.811(2)	0.888(1)	0.889(2)
hkchengrex	0.854(2)	0.828(3)	0.814(1)	0.883(3)	0.893(1)
testing-gg	0.854(3)	0.836(1)	0.806(3)	0.888(2)	0.885(3)
qinghualiuyong	0.842(4)	0.816(5)	0.799(4)	0.870(5)	0.881(4)
cncyww	0.839(5)	0.823(4)	0.788(7)	0.874(4)	0.871(6)
cheng321284	0.836(6)	0.809(8)	0.798(5)	0.859(8)	0.877(5)
PixelKitty	0.835(7)	0.814(6)	0.793(6)	0.866(6)	0.868(7)
dandan66	0.821(8)	0.811(7)	0.765(10)	0.864(7)	0.845(10)
JerryX	0.818(9)	0.795(9)	0.780(8)	0.845(9)	0.853(8)
BeyondID	0.810(10)	0.784(11)	0.770(9)	0.835(11)	0.849(9)
niceL	0.806(11)	0.786(10)	0.763(11)	0.836(10)	0.837(11)

Table 2. Ranking results in the YouTube-VOS 2021 test set. We mark our results in blue.

References

[1] M. Berman, A. Rannen Triki, and M. B. Blaschko. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4413–4421, 2018.

[2] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.

[3] X. Chen, B. Yan, J. Zhu, D. Wang, X. Yang, and H. Lu. Transformer tracking. In *CVPR*, 2021.

[4] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S.-M. Hu. Global contrast based salient region detection. *IEEE transactions on pattern analysis and machine intelligence*, 37(3):569–582, 2014.

[5] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[6] Y. T. Hu, J. B. Huang, and A. G. Schwing. Maskrnn: Instance level video object segmentation. *Advances in Neural Information Processing Systems*, 2017:325–334, 2017.

[7] Y.-T. Hu, J.-B. Huang, and A. G. Schwing. Videomatch: Matching based video object segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 54–70, 2018.

[8] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[9] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[10] S. W. Oh, J.-Y. Lee, N. Xu, and S. J. Kim. Video object segmentation using space-time memory networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9226–9235, 2019.

[11] F. Perazzi, A. Khoreva, R. Benenson, B. Schiele, and A. Sorkine-Hornung. Learning video object segmentation from static images. In *CVPR*, pages 3491–3500, 2017.

[12] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, and L. Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

[13] J. Shi, Q. Yan, L. Xu, and J. Jia. Hierarchical image saliency detection on extended cssd. *IEEE transactions on pattern analysis and machine intelligence*, 38(4):717–729, 2015.

[14] P. Voigtlaender, Y. Chai, F. Schroff, H. Adam, B. Leibe, and L.-C. 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.

[15] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. Torr. Fast online object tracking and segmentation: A unifying approach. In *CVPR*, pages 1328–1338, 2019.

[16] A. C. Wilson, R. Roelofs, M. Stern, N. Srebro, and B. Recht. The marginal value of adaptive gradient methods in machine learning. In *NIPS*, 2017.

[17] N. Xu, L. Yang, Y. Fan, J. Yang, D. Yue, Y. Liang, B. Price, S. Cohen, and T. Huang. Youtube-vos: Sequence-to-sequence video object segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 585–601, 2018.

[18] Z. Yang, Y. Wei, and Y. Yang. Collaborative video object segmentation by foreground-background integration. In *European Conference on Computer Vision*, pages 332–348. Springer, 2020.

[19] Z. Zhou, L. Ren, P. Xiong, Y. Ji, P. Wang, H. Fan, and S. Liu. Enhanced memory network for video segmentation. In *IEEE International Conference on Computer Vision Workshops*, pages 689–692, 2019.