# Supplementary Material of "Video Object Segmentation with Dynamic Memory Networks and Adaptive Object Alignment"

Shuxian Liang[1, 2]*, Xu Shen[2], Jianqiang Huang[2], Xian-Sheng Hua[2†]
[1] State Key Lab of CAD&CG, Zhejiang University, [2] DAMO Academy, Alibaba Group
shuxian.lsx@zju.edu.cn, {shenxu.sx,jianqiang.hjq,xiansheng.hxs}@alibaba-inc.com

In the supplementary material, we will provide more technical background, implementation details, experimental results and visualized results of our method. Specifically, we discuss the related works without matching in Sec. 1. We show the data augmentation approaches for training and the settings of our matching layer in Sec. 2. Moreover, we display our experimental results on DAVIS 2016 validation set in Sec. 3. we present the predicted masks before/after refinement in Sec. 4, visualization of the memories in Sec. 5, visualization of the alignment in Sec. 6 and some failure cases in Sec. 7.

## 1. Extra Related Works

**Online Learning.** Fine-tuning models towards target objects at test time is a simple yet effective solution to recognize one-shot objects in VOS. OSVOS [1] fine-tunes its segmentation network on labelled initial frames at test time. OnAVOS [14] fine-tunes extra on subsequent predicted frames with high output confidences. LucidTracker [3] enriches initial-frame data using data augmentation. PReMVOS [7] integrates several fine-tuned networks (instance segmentation, mask refinement, and object ReID), which makes it powerful but costly. More recently, FRTM [11] fine-tunes a lightweight object module and has it worked with an offline-trained segmentation network. Actually, the online fine-tuning step could generally bring performance improvements to the PVOS/OVOS methods (*e.g.* [4, 13, 17]). However, it is usually time-consuming so that not always feasible in real-world scenarios.

**Mask Propagation.** Mask propagation is the base of the popular PVOS/OVOS methods, which works by propagating either labelled or predicted masks for future references. However, early works using mask propagation do not involve matching. For instance, MaskTrack [9] inputs previous masks additionally to its segmentation network in order to refine segmentation. MaskRNN [2] propagates masks frame by frame through a recurrent neural network and the optical flow. RGMP [15] uses both first-frame and previous-frame information via a siamese encoder-decoder network. Notably, compared with the online learning, the mask propagation mechanism could be learned completely offline and much more applicable in practice . As a result, it is further developed to work with matching and extended to exploit as many as possible frames in many later works (*e.g.* [5, 6, 8]).

## 2. More Implementation Details

**Data Augmentation.** As mentioned in the main paper, we use video clips for training. For a given clip, firstly, all frames are rescaled with the same factor randomly sampled in $[1, 1.5]$. Secondly, these frames are flipped horizontally and simultaneously with a probability of $0.5$. Thirdly, these frames are flipped temporally (*i.e.* inverse the order of frames) with a probability of $0.5$. Lastly, input patches ($352 \times 352$) are randomly cropped from the augmented frames.

**Settings of The Matching Layer.** Our aligned matcher consists of the proposed adaptive object alignment module and a matching layer. The matching layer we use is proposed in [17]. It is differentiable and performs like the Hungarian algorithm. For both training and testing, $N_{grad}$, $N_{proj}$ and $\alpha$ are set to 10, 5 and 0.1, respectively. Notably, $\lambda$ is automatically tuned as a parameter for each dataset.

## 3. Results on DAVIS 2016

DAVIS 2016 [10] is an early and simplified version of DAVIS 2017, which contains 30 videos for training and 20 videos for validation. Each video in DAVIS 2016 has a single target object. Notably, the datasets in the main paper typically include individual objects (*e.g.* a 'person'/'car'/'dog') only, while this dataset additionally includes some compositional objects (*e.g.* the single 'soapbox' object consists of two 'persons' and a 'trolley').

Results of our method and the SOTA methods are presented in Table 1. Compared with the best reported performances on DAVIS 2016 validation set, our method achieves

*This work was done when the author was visiting Alibaba as a research intern.
†Corresponding author.

| Models | $\mathcal{J}$ | $\mathcal{F}$ | $\mathcal{G}$ |
|---|---|---|---|
| GC [5] (ECCV20) | 87.6 | 85.7 | 86.6 |
| PReMVOS [7] (ACCV18) | 84.9 | 88.6 | 86.8 |
| STM [8] (ICCV19) | 84.8 | 88.1 | 86.5 |
| STM (+YV) [8] (ICCV19) | 88.7 | 89.9 | 89.3 |
| CFBI [16] (ECCV20) | 85.3 | 86.9 | 86.1 |
| CFBI (+YV) [16] (ECCV20) | 88.3 | 90.5 | 89.4 |
| KMN [12] (ECCV20) | 87.1 | 88.1 | 87.6 |
| KMN (+YV) [12] (ECCV20) | 89.5 | 91.5 | 90.5 |
| Ours | 88.1 | 89.3 | 88.7 |
| Ours (YV) | **90.5** | **92.3** | **91.0** |

Table 1. Quantitative results on DAVIS 2016 validation set. $YV$ indicates an extra use of YTVOS for training

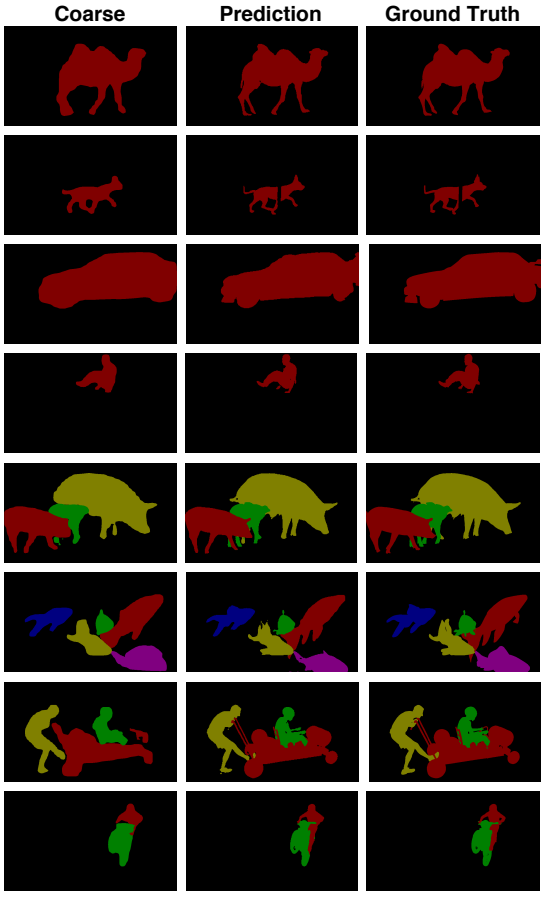


Figure 1. Predicted object masks of our method on DAVIS 2017 validation set. The top four rows are for single-object segmentation and the bottom four rows are for multi-object segmentation.

better results ($+1.1\%$ and $+0.5\%$) with or without an extra use of YTVOS. The results verify that our method could generalizes well to both the individual objects and the compositional objects.

## 4. Visualization of Object Masks

To validate the effectiveness of our method, here we show more qualitative results in Fig. 1. Specifically, other than the ground-truth masks, we display the coarse masks (before refinement) as well as the refined masks (after refinement). The top four rows are cases of single-object segmentation, covering objects of animals, vehicles and persons. The bottom four rows are cases of multi-object segmentation, covering the co-occurrence of persons, animals and vehicles.

As shown in the Fig 1, our coarse masks are already quite accurate on the locations, sizes and shapes of target objects. This reveals that our object matching results are highly reliable, thanks to the dynamic memory networks and the adaptive object alignment module. The *RefineNet*, on the other hand, further optimizes the segmentation quality by refining the details such as boundaries (all rows), occlusions (2nd, 5th, 6th, 7th and 8th rows) and missing parts (5th, 6th and 7th rows). To sum up, in our framework, the object matching and the *RefineNet* work collaboratively to achieve highly accurate video object segmentation.

## 5. Visualization of Dynamic Memory Networks

In Fig. 2, we visualize the aforementioned query-key relevance scores of the dynamic memory networks. Specifically, we first compute the relevance scores between every pixel of the space-time key feature maps from the memory frames and all object pixels of the query feature map from the current frame. Then, we visualize the normalized scores on the memory frames using heat map based visualization.

As presented in Fig. 2, the target object in the first frame presents incomplete object information with regard to the current frame due to the scale/view/pose/occlusion variations. Our method tackles this problem by learning the complementary object information from the memory frames. Specifically, for the 1st case, the target object in the first frame presents the lower half clues (trousers, waist, etc) of the person appearance only and the appearance of the upper half (clothes from 2nd frame, hands from 3rd frame, etc.) is obtained from the memory frames. For the 2nd case, the target object in the first frame provides the frontal appearance only and the dorsal appearance of the person is gathered from the memory frames. For the 3rd case, the target object in the first frame is coarsely represented due to its small scale and more details of the target are obtained from the memory frames. For the last case, the target object in the first frame misses the appearance of the flank and our model obtains this clue from the memory frames.

## 6. Visualization of Adaptive Object Alignment

In Fig. 3, we visualize the template-proposal matching with or without the proposed adaptive object alignment. For

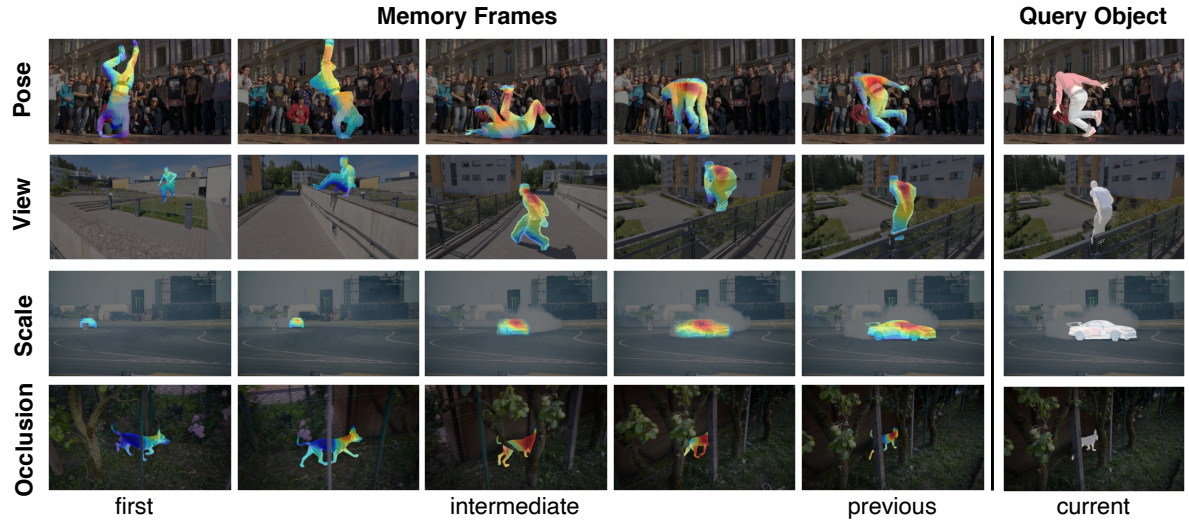


Figure 2. Visualization of the query-key relevance score heat maps of the dynamic memory networks. Specifically, the pixels in red have high scores, the pixels in green have medium scores and the pixels in blue have low scores. For all cases, the target object in the first frame presents incomplete object information with regard to the current frame due to scale/view/pose/occlusion variations. Our method tackles this problem by learning to obtain the complementary object information from the memory frames.


Figure 3. Visualization of the template-proposal matching with or without the adaptive object alignment. For the simplicity of the visualization, we first sample a few checkpoints (marked by a white dot) from the templates (the left column). Using the alignment (the middle column), each checkpoint is matched with the weighted average of *all* locations on a given proposal (only the highest weighted ones are shown here and straight arrows with different colors indicate different proposals). Without using the alignment (the right column), each checkpoint is matched with its spatially corresponding location on the given proposal instead.

the simplicity of visualization, we use the first-frame annotations as the templates instead of the dynamic templates. And we sample a few checkpoints from all given templates.

As shown in the the middle column, using the alignment, each checkpoint is matched with the weighted average of *all* locations on a given proposal. Weights are the similarity scores between each pixel of the proposal object and all pixels of the template in the feature space.

As can be observed in the visualization, the object matching without the alignment might matches the checkpoints with the semantically-unrelated locations, such as negative pixels (1st and 2nd case) and the positive pixels from the very different parts (all cases). The problem of template-proposal misalignment is well tackled by the proposed adaptive object alignment. With our method, the checkpoints are mainly matched with the positive pixels in the proposals. More importantly, the checkpoint of a distinguishable part of the target object is mainly matched with the positive pixels from the same semantic part in the proposals (the 1st, 2nd and 3rd rows). This reveals that the adaptive object alignment method could softly align proposals with templates in the feature space. In addition, the last row in Fig. 3 shows a hard case where the target object is severely occluded, resulting in multiple positive proposals (all contain positive pixels). For this case, the checkpoints are perfectly matched with the most similar positive pixels of every proposal by our model. In this way, compared to the matching without the alignment, our method compensates the information loss of the incomplete positive proposals and prevent the mismatches between parts of proposals and the template.

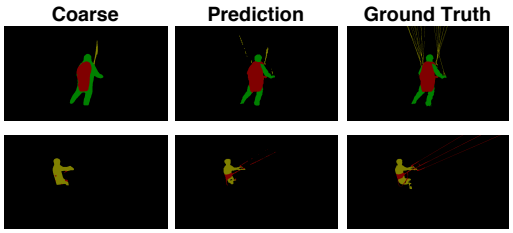


Figure 4. Failure cases of our method.

## 7. Failure Case Study

In Fig 4, we show two failure cases of our method. In both cases, the large-scale objects are well segmented from the background. However, for the small-scale objects (the strings in the 1st and 2nd rows), our method produces unsatisfactory results due to two main reasons. Firstly, using the high-level feature maps from the backbone network, the small-scale objects are possibly overlooked at some stages of the whole inference process, including the detection, the object matching and the final refinement step. Secondly, the bounding boxes of the oblique thin strings contain much more negative pixels with regard to the positive pixels, resulting in the noisy representations of the objects in both the detection and the matching stages. Notably, similar failure cases are also observed in many other SOTA methods like STM [8] and CFBI [16]. The segmentation of such small-scale objects is actually one of the very important but unsolved problems in semi-supervised video object segmentation.

## 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 221–230, 2017. 1

[2] Yuan-Ting Hu, Jia-Bin Huang, and Alexander G Schwing. Maskrnn: Instance level video object segmentation. *arXiv preprint arXiv:1803.11187*, 2018. 1

[3] 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. 1

[4] Xiaoxiao Li and Chen Change Loy. Video object segmentation with joint re-identification and attention-aware mask propagation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 90–105, 2018. 1

[5] Yu Li, Zhuoran Shen, and Ying Shan. Fast video object segmentation using the global context module. In *European Conference on Computer Vision*, pages 735–750. Springer, 2020. 1, 2

[6] Yongqing Liang, Xin Li, Navid Jafari, and Qin Chen. Video object segmentation with adaptive feature bank and uncertain-region refinement. *arXiv preprint arXiv:2010.07958*, 2020. 1

[7] Jonathon Luiten, Paul Voigtlaender, and Bastian Leibe. Premvos: Proposal-generation, refinement and merging for video object segmentation. In *Asian Conference on Computer Vision*, pages 565–580. Springer, 2018. 1, 2

[8] 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 International Conference on Computer Vision*, pages 9226–9235, 2019. 1, 2, 4

[9] Federico Perazzi, Anna Khoreva, Rodrigo Benenson, Bernt Schiele, and Alexander Sorkine-Hornung. Learning video object segmentation from static images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2663–2672, 2017. 1

[10] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 724–732, 2016. 1

[11] Andreas Robinson, Felix Jaremo Lawin, Martin Danelljan, Fahad Shahbaz Khan, and Michael Felsberg. Learning fast and robust target models for video object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7406–7415, 2020. 1

[12] 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. 2

[13] Jae Shin Yoon, Francois Rameau, Junsik Kim, Seokju Lee, Seunghak Shin, and In So Kweon. Pixel-level matching for video object segmentation using convolutional neural networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2167–2176, 2017. 1

[14] Paul Voigtlaender and Bastian Leibe. Online adaptation of convolutional neural networks for video object segmentation. In Gabriel Brostow Tae-Kyun Kim, Stefanos Zafeiriou and Krystian Mikolajczyk, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 116.1–116.13. BMVA Press, September 2017. 1

[15] 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. 1

[16] Zongxin Yang, Yunchao Wei, and Yi Yang. Collaborative video object segmentation by foreground-background integration. In *Proceedings of the European Conference on Computer Vision*, 2020. 2, 4

[17] Xiaohui Zeng, Renjie Liao, Li Gu, Yuwen Xiong, Sanja Fidler, and Raquel Urtasun. Dmm-net: Differentiable mask-matching network for video object segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3929–3938, 2019. 1