Towards Multi-Object Association from Foreground-Background Integration

Zongxin Yang[1,4], Jian Zhang[2], Wenhao Wang[1,3], Wenhua Han[2], Yue Yu[2],
Yingying Li[2], Jian Wang[2], Yunchao Wei[3], Yifan Sun[1], Yi Yang[4]
[1] Baidu Research [2] Baidu Inc.
[3] ReLER, University of Technology Sydney
[4] CCAI, College of Computer Science and Technology, Zhejiang University
{zongxinyang1996, wangwenhao0716, wychao1987, yee.i.yang}@gmail.com
{zhangjian52, hanwenhua, yuyue15, liyingying05, wangjian33, sunyifan01}@baidu.com

Abstract

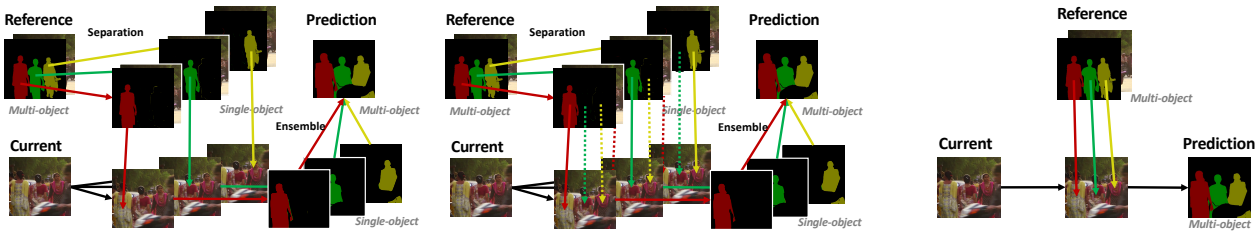
This paper investigates how to realize better and more efficient embedding learning to tackle the semi-supervised video object segmentation under challenging multi-object scenarios. Although one of the state-of-the-art VOS methods, CFBI, has achieved promising performance by integrating foreground-background information, each positive object is decoded individually under multi-object scenarios. To associate all video objects, we propose an Associating Objects with Transformers (AOT) [25] approach to match and decode multiple objects uniformly. In detail, AOT employs an identification mechanism to associate multiple targets into the same high-dimensional embedding space. Thus, we can simultaneously process the matching and segmentation decoding of multiple objects as efficiently as processing a single object. For sufficiently modeling multi-object association, a Long Short-Term Transformer is designed for constructing hierarchical matching and propagation. We conduct extensive experiments on YouTube-VOS to examine AOT variant networks with different complexities. Compared to CFBI+ (82.8%, 4.0FPS), our AOT-S (82.6%, 12.5FPS) achieves comparable accuracy and 3× speed on the validation 2018 split. Our larger variant, AOT-L (83.7%, 6.3FPS), achieves superior performance even using a light-weight backbone, MobileNet-V2. After applying test-time augmentations and model ensemble, we ranked $\mathbf{1^{st}}$ *in Track 1 (Video Object Segmentation) of the 3rd Large-scale Video Object Segmentation Challenge. The code will be publicly available at https://github.com/z-x-yang/AOT.*

1. Introduction

Thanks to the recent advance of deep neural networks, many deep learning based video object segmentation (VOS) algorithms have been proposed recently and achieved promising performance. STM [15] and its following works [17, 14] leverage a memory network to store and read the target features of predicted past frames and apply a non-local attention mechanism to match the target in the current frame. FEELVOS [20] and CFBI [24, 26] utilize global and local matching mechanisms to match target pixels or patches from both the first and the previous frames to the current frame. Particularly, CFBI proposed to integrate both the foreground and background features to learn contextual information and contrastive object embeddings. Such a simple foreground-background integration has shown promising improvement and indicates that contextual information is important for effective embedding learning.

Even though the above methods have achieved significant progress, the above methods learn to decode scene features that contain a single positive object. Thus under a multi-object scenario, they have to match each object independently and ensemble all the single-object predictions into a multi-object segmentation, as shown in Fig. 1a. Such a post-ensemble manner eases network architectures' design since the networks are not required to adapt the parameters or structures for different object numbers. However, processing multiple objects separately yet in parallel requires multiple times the amount of GPU memory and computation for processing a single object. This problem restricts the training and application of VOS under multi-object scenarios, especially when computing resources are limited. Besides, modeling multiple objects independently, instead of uniformly, is inefficient in exploring multi-object contextual information, which should be important to learn more robust feature embeddings, as motivated by the foreground-background integration.

To solve the above problems, Fig. 1c demonstrates a feasible approach to associate and decode multiple objects uniformly in an end-to-end framework. Hence, we pro-



(a) Post-ensemble (foreground-only) (b) Foreground-background Integration (c) Associating Objects

Figure 1: (a) Many VOS methods process multi-object scenarios in a post-ensemble manner. (b) CFBI [24, 26] introduced the foreground-background integration by additionally matching the relative background for each object (dot lines). (c) Instead of using post-ensemble, Our AOT [25] associates all the objects in a end-to-end network, leading to better efficiency and embedding learning.

pose an Associating Objects with Transformers (AOT) [25] approach to match and decode multiple targets uniformly. First, an identification mechanism is proposed to assign each target a unique identity and embed multiple targets into the same feature space. Hence, the network can learn the association or correlation among all the targets. Moreover, the multi-object segmentation can be directly decoded by utilizing assigned identity information. Second, a Long Short-Term Transformer (LSTT) is designed for constructing hierarchical object matching and propagation. Each LSTT block utilizes a long-term attention for matching with the first frame's embedding and a short-term attention for matching with several nearby frames' embeddings. Compared to the methods [15, 17] utilizing only one attention layer, we found hierarchical attention structures are more effective in associating multiple objects.

We conduct extensive experiments on YouTube-VOS [23] to validate the effectiveness and efficiency of the proposed AOT. Even using the light-weight MobileNet-V2 [16] as the backbone encoder, the AOT variant networks achieve superior performance on the validation 2018 & 2019 splits of the large-scale YouTube-VOS (ours, $\mathcal{J}\&\mathcal{F}$ **82.6**$\sim$**83.7**% & **82.2**$\sim$**83.6**%) while keeping faster multi-object run-time (**12.5**$\sim$**6.3**FPS) compared to the state-of-the-art competitors (*e.g.*, CFBI [24], 81.4% & 81.0%, 3.4FPS). Besides, our smallest variant, AOT-T, can maintain real-time multi-object speed on YouTube-VOS. After applying common test-time augmentations (multi-scale and flipping) and ensembling AOT-L [25] with CFBI+ [26], we ranked $\mathbf{1^{st}}$ in the Track 1 (Video Object Segmentation) of the 3rd Large-scale Video Object Segmentation Challenge.

2. Revisit Foreground-Background Integration

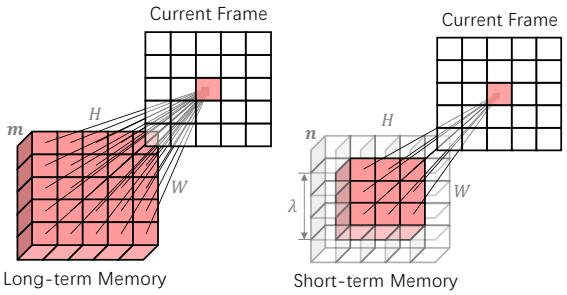
Benefit from deep networks, current state-of-the-art VOS methods [15, 20] have achieved promising performance. Nevertheless, these methods focus on matching and decoding a single object. Under a multi-object scenario, they thus have to match each object independently and ensemble all the single-object predictions into a multi-object prediction, as demonstrated in Fig. 1a. This manner extends networks designed for single-object VOS into multi-object applications, so there is no need to adapt the network for different object numbers.

Based on such a post-ensemble manner, CFBI [24, 26] introduced the concept of foreground-background integration, *i.e.*, additionally matching the relative background for each object, as shown in Fig. 1b. Compared to foreground-only matching, the foreground-background integration leverages more contextual information and thus can relieve the background confusion problem [24], leading to more accurate segmentation.

Although the above post-ensemble manner is prevalent and straightforward in the VOS field, processing multiple objects separately yet in parallel requires multiple times the amount of GPU memory and computation for matching a single object and decoding the segmentation. This problem restricts the training and application of VOS under multi-object scenarios when computing resources are limited. To make the multi-object training and inference as efficient as single-object ones, an expected solution should be capable of associating and decoding multiple objects uniformly instead of individually (Fig. 1c). To achieve such an objective, we propose the AOT framework to associate and segment multiple objects uniformly within an end-to-end framework, leading to better efficiency. Compared to foreground-background integration, our training is more efficient since AOT can associate multiple object regions and learn comprehensive contextual information directly.

3. Associating Objects with Transformers

This section introduces the AOT framework, including our identification mechanism proposed for efficient multi-object VOS and the long short-term transformer for constructing hierarchical multi-object matching and propaga-



(a) Long-term Attention (b) Short-term Attention

Figure 2: Illustrations of the long-term attention and the short-term attention.

tion. More details can be found in [25].

3.1. Identification Mechanism

We propose an identification mechanism consisting of identification embedding and decoding based on attention mechanisms to associate multiple objects.

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

For **Identification Decoding**, *i.e.*, predicting all the targets' probabilities from the aggregated feature, we firstly predict the probability logit for every identity in the bank by employing a convolutional decoding network, and then select the assigned ones and calculate the probabilities.

3.2. Long Short-Term Transformer

Recently, transformer blocks [19] have been demonstrated to be stable and promising in constructing hierarchical attention structures in visual tasks [1, 6]. We carefully design a Long Short-Term Transformer (LSTT) block for multi-object VOS based on transformer blocks.

Following the common transformer blocks [19, 5], LSTT firstly employs a self-attention layer, which is responsible for learning the association or correlation among the targets within the current frame. Then, LSTT additionally introduces a long-term attention (Fig. 2a), for aggregating targets' information from long-term memory frames and a short-term attention (Fig. 2b), for learning temporal smoothness from nearby short-term frames. The final module is a common 2-layer feed-forward MLP with GELU [10] non-linearity in between.

The hierarchical matching and propagation of LSTT are not simply a stack of multiple attention processes. 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 [25].

4. Implementation Details

We follow the original setting of AOT [25] to build AOT variants, including AOT-T, AOT-S, AOT-B, and AOT-L. In the default setting, only a light-weight network, MObileNet-V2 [16], is used as the backbone encoder.

All the training details are the same as the strategy used in AOT [25], where the training stage is divided into two phases: (1) pre-training on synthetic video sequence generated from static image datasets [7, 13, 4, 18, 8] by randomly applying multiple image augmentations [22]. (2) main training on the VOS benchmarks [23] by randomly applying video augmentations [24].

We evaluate our AOT on YouTube-VOS [23], which is the latest large-scale benchmark for multi-object video segmentation. Specifically, YouTube-VOS contains 3471 videos in the training split with 65 categories and 474/507 videos in the validation 2018/2019 split with additional 26 unseen categories. The unseen categories do not exist in the training split in order to evaluate the generalization ability of algorithms.

When evaluating, all the videos are restricted to be not bigger than $1.3 \times 480p$ resolution [24, 26, 25]. When using multi-scale test-time augmentation, the scales are $\{0.75\times, 1.0\times, 1.25\times, 1.5\times\}$.

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

5. The 3rd YouTube-VOS Challenge

In this section, we introduce our solution on The 3rd YouTube-VOS Challenge. We mainly adopt two frameworks, AOT and CFBI+. AOT is a newly proposed transformer-based method which is elaborated above. Apart from AOT, we also enhance CFBI+ for further improvement. With the strength of model ensembling, we finally achieve the $\mathbf{1^{st}}$ rank on the test split of this challenge.

Table 1: The quantitative evaluation of AOT [25] on YouTube-VOS [23]. For sufficiently validating the effectiveness, all the AOT models use light-weight MobileNet-V2 [16] as the backbone encoder. ^{MS}: using a multi-scale and flipping strategy during inference.

		Seen		Unseen		
Method	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	FPS
Validation 2018 Split						
STM [15]	79.4	79.7	84.2	72.8	80.9	-
KMN [17]	81.4	81.4	85.6	75.3	83.3	-
CFBI [24]	81.4	81.1	85.8	75.3	83.4	3.4
CFBI+ [26]	82.8	81.8	86.6	77.1	85.6	4.0
AOT-T	80.2	80.1	84.5	74.0	82.2	**25.3**
AOT-S	82.6	82.0	86.7	76.6	85.0	12.5
AOT-B	83.2	**82.6**	87.4	77.3	85.6	8.0
AOT-L	**83.7**	82.5	**87.5**	**77.9**	**86.7**	6.3
Validation 2019 Split						
CFBI [24]	81.0	80.6	85.1	75.2	83.0	3.4
CFBI+ [26]	82.6	81.7	86.2	77.1	85.2	4.0
AOT-T	79.7	79.6	83.8	73.7	81.8	**25.3**
AOT-S	82.2	81.3	85.9	76.6	84.9	12.5
AOT-B	83.3	82.5	87.0	77.8	86.0	8.0
AOT-L	83.6	82.2	86.9	78.3	86.9	6.3
AOT-L^{MS}	**84.6**	**83.8**	**88.4**	**79.0**	**87.1**	-

5.1. Compare AOT with SOTA methods

As shown in Table 1, AOT variants achieve superior performance on YouTube-VOS compared to the previous state-of-the-art methods. With our identification mechanism, AOT-S (82.6% $\mathcal{J}\&\mathcal{F}$) surpasses CFBI [24] (81.4%) by +1.2% while running about $4\times$ faster (12.5 *vs* 3.4FPS). By using more LSTT blocks, AOT-B effectively improves the performance to 83.2%. Moreover, by utilizing the memory reading strategy, the unseen scores of AOT can be further improved, and our AOT-L (**83.7%/83.6%**, 6.3FPS) significantly outperforms the previous methods (*e.g.*, CFBI, 81.4%/81.0%, 3.4FPS) on the validation 2018/2019 split while maintains an efficient speed. After applying test augmentations, we can further boost the performance of AOT-L to **84.6%** on the validation 2019 split. More comparisons can be found in [25].

5.2. Enhanced CFBI+

We enhance the CFBI+ from two aspects described below. Firstly, we develop several new feature extractors for stronger image-level feature representation. Secondly, a modified training strategy from the original CFBI+ is used for better performance. Table 2 illustrates the performance of Enhanced CFBI+.

Table 2: Performance of Enhanced CFBI+ on Youtube-VOS Validation 2019 split. The test-time augmentations are used during inference.

		Seen		Unseen	
Method	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$
Baseline	83.5	82.8	87.0	78.3	86.0
CFBI+(U-HRNet)	84.5	83.0	87.6	**80.0**	87.5
CFBI+(SFNet)	**84.8**	**83.5**	**88.0**	**80.0**	**87.9**

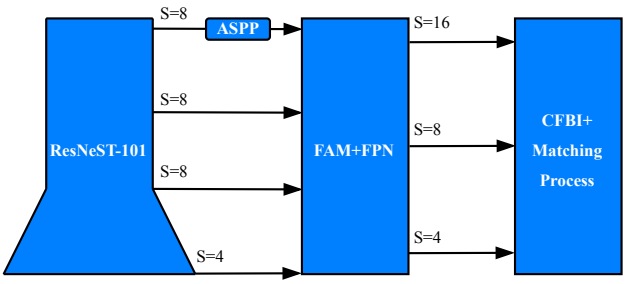


Figure 3: Structure of SFNet-CFBI+.

5.2.1 Feature Extractors

SFNet: In order to enhance the semantic representation of the output of the feature extractor, we choose a more powerful backbone, ResNeSt-101 [27], with the ASPP module [2], as demonstrated in Fig. 3. The ASPP module helps us capture the contextual information at multiple scales. Furthermore, we use Feature Pyramid Network (FPN) [12] to fuse the information from small scales to large scales. Meanwhile, to align feature maps of two adjacent levels in a feature pyramid, we make use of a Flow Alignment Module (FAM) [11] here.

When it comes to the network details, we first extract four feature maps with different strides ($S = 4, 8, 8, 8$) from the backbone and apply the ASPP module to the highest level feature map. Then the output of the ASPP and the other three feature maps were sent to the fusion module, consisting of the Flow Alignment Module (FAM) and Feature Pyramid Network (FPN). Moreover, the fusion module will generate three enhanced feature maps with different strides ($S = 4, 8, 16$). Finally, the enhanced feature maps were sent to the CFBI+ network to do the matching process.
U-HRNet: For semantic segmentation tasks, the strength of high-resolution representation is also crucial for performance. Here, we apply a high-resolution network named U-HRNet, as shown in Fig. 4. It inherits the advantages of HRNet [21], presented as maintaining high-resolution branches in parallel and performing multi-scale fusion throughout the network. Meanwhile, it further improves the semantic representation and propagates the strongest semantic representation to the highest resolution more effectively.

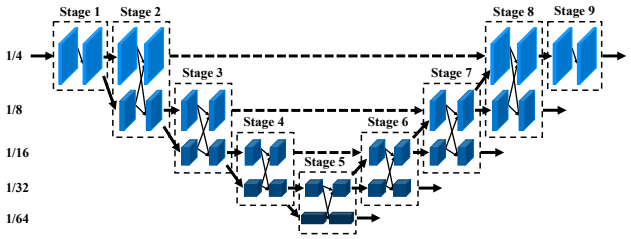


Figure 4: Structure of U-HRNet.

In analogy to the SFNet above, when associating with CFBI+, three feature maps with strides of 4,8,16 are outputted from U-HRNet for computing multi-scale distance maps.

5.2.2 Training Strategy

In order to get better performance and reduce the training time consumption of CFBI+, we pretrained SFNet and UHRNet with coco stuff which has 172 classes in advance. During the CFBI+ training, we use 8 Tesla V100 GPUs, and the batch size is 16. We use the initial learning rate of 2×10^{-2} for 50,000 steps. Then we will fine-tune the CFBI+ for 20,000 steps.

5.3. Model Ensembling

In the challenge, we apply an online model ensemble strategy, *i.e.*, the predictions from multiple models are ensembled frame by frame during the inference. In this way, we can use the ensembled better predictions as the memory masks dynamically, resulting in better performance compared to the common offline ensemble strategy.

5.4. Challenge Results

On the test split of this challenge, our best result ranked $\mathbf{1^{st}}$, which utilized 7 models in total, consisting of 3 frameworks, including 3 AOT-L [25] models, 3 Enhanced CFBI+ [26] models, and 1 KMN [15] model with Top-K attention [3]. The models which share the same framework diverse in different backbones ([27, 16, 9] or [2, 11, 21]). As shown in Table 3, our solution achieved the best performance on the overall and seen scores.

6. Conclusion

We propose a novel and efficient approach for video object segmentation by associating objects with transformers (AOT) [25], which achieves superior performance and efficiency on YouTube-VOS. A simple yet effective identification mechanism is proposed to associate, match, and decode all the objects uniformly under multi-object scenarios. In addition, a long short-term transformer is designed for constructing hierarchical object matching and propagation for VOS. The hierarchical structure allows us to flexibly balance AOT between real-time speed and state-of-the-art performance by adjusting the layer number. After ensembling two frameworks, AOT [25] and CFBI+ [26], we ranked $\mathbf{1^{st}}$ in the Track 1 (Video Object Segmentation) of the 3rd Large-scale Video Object Segmentation Challenge. We hope both of the frameworks will serve as solid baselines for VOS, and the more efficient AOT will help ease the future study of multi-object VOS and related tasks (*e.g.*, video instance segmentation, interactive video object segmentation, and multi-object tracking).

Table 3: The leaderboard of the VOS Challenge.

		Seen		Unseen	
Team	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$
cheng321284	83.6	80.9	85.9	79.8	87.7
cncyww	83.9	82.3	87.4	78.8	87.1
qinghualiuyong	84.2	81.6	87.0	79.9	88.1
testing-gg	85.4	**83.6**	**88.8**	80.6	88.5
hkchengrex	85.4	82.8	88.3	**81.4**	**89.3**
Ours	**85.6**	**83.6**	**88.8**	81.1	88.9

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, pages 213–229. Springer, 2020. 3

[2] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, pages 801–818, 2018. 4, 5

[3] Ho Kei Cheng, Yu-Wing Tai, and Chi-Keung Tang. Modular interactive video object segmentation: Interaction-to-mask, propagation and difference-aware fusion. In *CVPR*, 2021. 5

[4] Ming-Ming Cheng, Niloy J Mitra, Xiaolei Huang, Philip HS Torr, and Shi-Min Hu. Global contrast based salient region detection. *TPAMI*, 37(3):569–582, 2014. 3

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171—-4186, 2019. 3

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 3

[7] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010. 3

[8] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *ICCV*, pages 991–998. IEEE, 2011. 3

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[10] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 3

[11] Junghyup Lee, Dohyung Kim, Jean Ponce, and Bumsub Ham. Sfnet: Learning object-aware semantic correspondence. In *CVPR*, pages 2278–2287, 2019. 4, 5

[12] 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. 4

[13] 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 *ECCV*, pages 740–755. Springer, 2014. 3

[14] Xinkai Lu, Wenguan Wang, Martin Danelljan, Tianfei Zhou, Jianbing Shen, and Luc Van Gool. Video object segmentation with episodic graph memory networks. In *ECCV*, 2020. 1

[15] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. In *ICCV*, 2019. 1, 2, 4, 5

[16] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR*, pages 4510–4520, 2018. 2, 3, 4, 5

[17] Hongje Seong, Junhyuk Hyun, and Euntai Kim. Kernelized memory network for video object segmentation. In *ECCV*, 2020. 1, 2, 4

[18] Jianping Shi, Qiong Yan, Li Xu, and Jiaya Jia. Hierarchical image saliency detection on extended cssd. *TPAMI*, 38(4):717–729, 2015. 3

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017. 3

[20] Paul Voigtlaender, Yuning Chai, Florian Schroff, Hartwig Adam, Bastian Leibe, and Liang-Chieh Chen. Feelvos: Fast end-to-end embedding learning for video object segmentation. In *CVPR*, pages 9481–9490, 2019. 1, 2

[21] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 4, 5

[22] Seoung Wug Oh, Joon-Young Lee, Kalyan Sunkavalli, and Seon Joo Kim. Fast video object segmentation by reference-guided mask propagation. In *CVPR*, pages 7376–7385, 2018. 3

[23] 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. 2, 3, 4

[24] Zongxin Yang, Yunchao Wei, and Yi Yang. Collaborative video object segmentation by foreground-background integration. In *ECCV*, 2020. 1, 2, 3, 4

[25] Zongxin Yang, Yunchao Wei, and Yi Yang. Associating objects with transformers for video object segmentation, 2021. 1, 2, 3, 4, 5

[26] Zongxin Yang, Yunchao Wei, and Yi Yang. Collaborative video object segmentation by multi-scale foreground-background integration. *TPAMI*, 2021. 1, 2, 3, 4, 5

[27] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Zhi Zhang, Haibin Lin, Yue Sun, Tong He, Jonas Muller, R. Manmatha, Mu Li, and Alexander Smola. Resnest: Split-attention networks. *arXiv preprint arXiv:2004.08955*, 2020. 4, 5