Bi-Directional Seed Attention Network for Interactive Image Segmentation

Gwangmo Song and Kyoung Mu Lee

***Abstract*—In interactive segmentation, the role of seed information provided by the user is significant. A seed is a clue to ease the ambiguity of the problem by making the object segmentation task interactive. However, in most deep network-based works, seed information has been used as an additional channel for input images. In this paper, we propose a novel bi-directional attention module for more actively using seed information. The proposed bi-directional seed attention module (BSA) operates based on the feature map of the segmentation network and the input seed map. Through our attention module, the network feature map is affected by the seed map, while the feature also updates the seed information. As a result, our system concentrates on the seed information and more accurately derives the segmentation result required by the user. We have conducted validation experiments on the four standard benchmark datasets, including SBD, GrabCut, Berkeley, and DAVIS, and achieved the state-of-the-art results.**

***Index Terms*—Attention, deep network, interactive segmentation.**

I. Introduction

DUE to the recent development of deep learning, many computer vision fields have made significant progress. Meanwhile, an essential element for the development of such deep learning-based algorithms is the presence of a lot of accurate data. As a tool for labeling data more efficiently, interactive segmentation has recently been spotlighted. Interactive segmentation receives information from the annotator about the object to be labeled and outputs the segmentation result. The most crucial component of interactive segmentation is the seed information provided by the user. Since it is the only information that contains the user's intention, how to use the seed information directly affects the quality of the segmentation mask. Therefore, dealing seed information in a deep network is a significant issue.

Recent deep learning-based interactive segmentation algorithms [1]–[4] show remarkable performance improvement. Most of these algorithms operate based on seed points given in click form. The seed click information is converted into a distance map and used as an additional channel of the input image. However, in this case, the seed information may be weakened while passing through the deep layer, which leads to inaccurate segmentation results. Recently, BRS algorithms [5], [6] have noted the problem of forgetting seed information. They transformed the initial seed information or network features so that the seed information matched with the final label results. However, they have the problem of repeating forward and backward steps of the network several times. In contrast, we propose a method of deriving an accurate segmentation mask by repeatedly using seed information in a single forward step.

Seed indicates label information of the object located at the point. That is, the seed contains spatial information about a part of GT. Meanwhile, since the segmentation network is composed of a fully convolutional structure, spatial information is preserved through the network. Therefore, the label information of the input seed should be delivered to the label information of the final output mask without losing its spatial information. In order to preserve the seed information and transmit it to the output, the seed information must be emphasized not only at the input but also at an intermediate stage of the network. Maintaining the seed information in the network helps to create a better segmentation mask by strengthening the semantic information of objects around the seed.

We used the attention module to utilize the seed information. The attention mechanism works to strengthen the context information of the feature by making the network focus on the critical part. In the case of interactive segmentation, a seed can provide information on where the critical part is. In this paper, we proposed a bi-directional attention module and newly applied it to the interactive segmentation problem. We call our module Bi-directional Seed Attention (BSA). Through bi-directional attention, the feature map of the network pays attention to the seed map and accepts its spatial information. At the same time, the seed map focuses on the semantic information of the feature and changes to contain more relevant information. In other words, the feature map with strong semantic information and the seed map with strong spatial information exchange information with each other to improve the segmentation results.

We compared the results with existing interactive segmentation models. We trained using SBD dataset [7], which is widely used in segmentation tasks, and compared the number of clicks required to reach a target accuracy. In addition to the SBD dataset, we experiment on the GrabCut [8], Berkeley [9], and DAVIS [10] datasets, and the proposed algorithm recorded state-of-the-art results in comparison with various algorithms.

Manuscript received July 2, 2020; revised August 15, 2020; accepted August 18, 2020. Date of publication August 27, 2020; date of current version September 10, 2020. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Xun Cao. *(Corresponding author: Kyoung Mu Lee.)*

The authors are with the Department of Electrical Engineering and Computer Science, Automation and Systems Research Institute, Seoul National University, Seoul 151-600, South Korea (e-mail: kfsgm@snu.ac.kr; kyoungmu@snu.ac.kr).

This letter has supplementary downloadable material available at https://ieeexplore.ieee.org, provided by the authors.

Digital Object Identifier 10.1109/LSP.2020.3019970

II. Related Work

Interactive Segmentation. Earlier works like GrabCut [8], Graph Cut [11], Random Walks [12], RWR [13], GSC [14] tried to solve the problem by formulating and optimizing MRF models.

The first work that applied the deep learning frame to the interactive segmentation problem is DIOS [1]. The FCN [15] network structure that solved the semantic segmentation problem was modified and used. RIS-Net [3] improves segmentation accuracy by separating global and local branches. LD [2] increased diversity by focusing on the ambiguity of segmentation and suggesting multiple possible masks from a single input. In the FCTSFN [16], they did not combine RGB images and seed information at the input, but proceeded to each network stream and then fused. CMG [4] has improved performance by additionally using a guidance map containing context information as input. MultiSeg [17] improved their original DIOS method and designed a network that produces various segmentation results according to scale.

On the other hand, BRS [5] applied a backpropagating refinement scheme to solve the difference between the seed label information and the corresponding point label of the predicted mask. They gave perturbation to the input seed map so that the object mask result matches the input seed information. However, BRS takes a long time because it has to repeat the network several times until the condition is satisfied. To solve this, fBRS [6] significantly reduced the time by giving perturbation at the feature level without giving perturbation to the input. These BRS-based techniques are similar to our work in that they aim to maintain seed information through the network.

Attention Mechanism. Recently, the attention model has been applied to the vision field, resulting in high-performance improvement [18]–[22]. In particular, it was successfully applied to semantic segmentation ([23]–[25]), similar to the problem we are dealing with. These algorithms have in common that they apply the attention module to the feature map obtained through the backbone network. On the other hand, like BAM [26] and CBAM [27], we applied the attention module in the encoder network.

Li *et al* [28] have introduced a dual branch, as in our paper. They applied the attention model to the video salient object detection problem. They used an optical flow map to give attention information to the leading network. Unlike the [28], where additional parameters are significantly increased by using a separate network, we constructed the network efficiently using the newly proposed bi-directional attention module. Bi-directional attention strengthens the information of each element by paying attention to each other like [29], [30].

III. Proposed Method

A. Interactive Segmentation Network

Figure 1 shows the overall structure of our segmentation network. It consists of two parts; the backbone segmentation module and the BSA (Bi-directional Seed Attention) module. As in [1], the input of the network is the RGB image and seed map, and the output becomes the corresponding binary segmentation mask. Any segmentation network can be used as the backbone module. In our work, we used the structure in [6] that is based on DeeplabV3+ [31] as our backbone since it shows excellent performance. It is composed of the encoder, Atrous Spatial Pyramid Pooling (ASPP) module, and decoder. We used a structure that transmits low level features to decoder through skip connection to preserve local information, as in DeeplabV3+. The backbone encoder network is pretrained, and the seed map is obtained through distance transform as commonly done in other works.

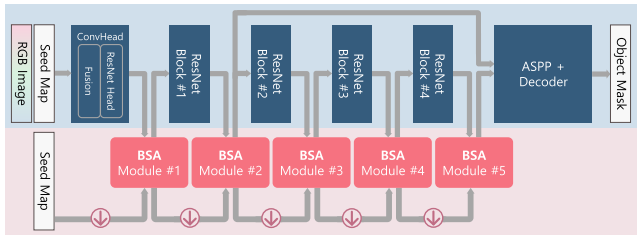

Fig. 1. Our interactive segmentation network architecture. The blue shaded part is the baseline network responsible for segmentation, and the red shaded part is the seed branch containing the newly proposed attention module. The downward purple arrow indicates the downscale operation.

ConvHead block located at the beginning of the network consists of two parts. One is the distance maps fusion module used in [6] to combine the RGB image and the seed map, and the other is the head module of ResNet [32]. The feature map created from the ConvHead block goes through the ResBlocks and decoder and produces the final result. However, as the seed information of the input passes deep through layers, it becomes difficult to maintain the seed information stable. Therefore, we employ attention modules to bring the seed information stably to the end of the network. As in Fig. 1, the output feature map of each ResBlock is not fed into the next block directly but after being updated through the BSA module. In the BSA module, the feature map is strengthened by the seed information to emphasize the semantic information of objects around the seed.

Meanwhile, the seed map also undergoes an update process. After going through the downscale process to fit the size with the feature map, it is updated based on the semantic information of the feature map. At this time, seed information is a kind of auxiliary variable. Therefore, instead of applying a separate loss function to the seed map, the seed map is converted to have the appropriate information to update the feature map. As shown in Fig. 2, the feature map of the baseline (backbone) network does not catch semantic information around the seed due to weakened information. On the other hand, the proposed BSA network preserves the seed information well enough through the layers, and in turn, makes the feature maps emphasize the semantic information about objects around the seed. The feature map shown in Fig. 2 is the output of the encoder and shows the average value of the channel dimension. The updated seed map shows the seed used in the last BSA module.

B. Bi-Directional Seed Attention Module

The structure of the proposed BSA module is shown in Fig. 3. Since it is a bi-directional structure, it has two inputs and two

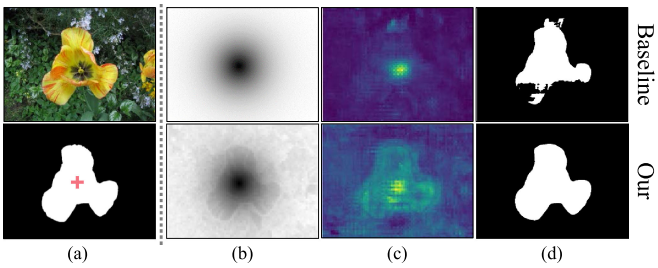

Fig. 2. Segmentation examples. (a) RGB input image, GT object mask and seed location (b) initial foreground seed distance map (upper) and updated foreground seed map (lower) (c) feature map of baseline (upper) and our (lower) (d) segmentation mask of the baseline (upper) and our BSA (lower).

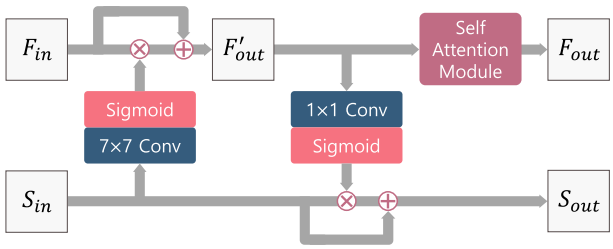

Fig. 3. Our BSA module. Both the multiplication and addition marks are element-wise operations. F'_{out} is used for module #1 and #5, and F_{out} is used for the remaining modules.

outputs. The feature map $F_{in} \in \mathbb{R}^{C\times H\times W}$ obtained from the ResBlock of the previous layer and the seed map $S_{in} \in \mathbb{R}^{2\times H\times W}$ of the seed branch are the input of our module, and the feature map $F_{out} \in \mathbb{R}^{C\times H\times W}$ to be used as input to the next layer and the updated seed map $S_{out} \in \mathbb{R}^{2\times H\times W}$ are output. The two channels of S_{in} and S_{out} represent foreground and background, respectively. We can mainly divide the operation of the module into three parts. Those are the part that updates the feature map based on the seed information, the part that updates the seed from the feature information, and the part that performs self-attention on the feature map.

In the bi-directional attention module, the feature map is updated first. For this, the seed map is converted into an attention map $A_S \in \mathbb{R}^{1\times H\times W}$ through a convolution operation, and then represented in the form of probability by a Sigmoid transformation. The attention map exhibits information about where to pay attention based on the input seed information. The attention map is then applied to the feature map through element-wise multiplication for each channel. Finally, we complete the feature map update through the residual operation. The seed update process goes through a similar fashion, as shown in Fig. 3. Both feature and seed map update processes can be described as follows.

$$F'_{out} = F_{in} + F_{in} \otimes \sigma(h^{7\times 7}(S_{in})), \tag{1}$$

$$S_{out} = S_{in} + S_{in} \otimes \sigma(h^{1\times 1}(F'_{out})), \tag{2}$$

where σ is the Sigmoid function and $\otimes$ means element-wise multiplication. The convolution operation is represented by $h^{k\times k}$, which has a kernel size of $k \times k$, and if necessary, preserves the size of the input through padding. The output channel size of $h^{k\times k}$ is 1, which serves as a channel reduction. For the feature update, we use a larger size kernel since the spatial information is crucial for the feature. In the case of the feature map, channel reduction is sufficient because the information to be transmitted to the seed is semantic information. However, in the case of a seed map, it is necessary to have a receptive field so that spatial information of the foreground seed and the background seed can be synthesized and converted into information suitable for the update.

Unlike the seed update, which directly outputs S_{out}, feature update takes one more step. We further concentrate the semantic information by using the self-attention module that uses its feature information rather than seed information. At this time, any module can be used for self-attention, and we employed BAM [26]. We obtain the final $F_{out} \in \mathbb{R}^{C\times H\times W}$ through BAM composed of spatial attention and channel attention. However, we did not adjust self-attention for all BSA modules, but only for modules #2, #3, and #4. It is according to the configuration in the original implementation of BAM. In module #1 and #5, F'_{out} was used instead of F_{out}. The following is a summary of our modules.

$$F_{out} = BAM(F_{in} + F_{in} \otimes \sigma(h^{7\times 7}(S_{in}))), \tag{3}$$

$$S_{out} = S_{in} + S_{in} \otimes \sigma(h^{1\times 1}(F_{in} + F_{in} \otimes \sigma(h^{7\times 7}(S_{in})))). \tag{4}$$

C. Implementation Details

We used ResNet-101 model as our backbone (baseline) networks for the experiment. We trained the networks using the SBD dataset [7] consisting of a total of 8,498 images. All images were cropped to be the same size of 320×480. For augmentation, we adopted random flipping and resizing. We used binary cross entropy loss for network optimization. During the first 100 epoch, a learning rate of 5×10^{-4} was used, and during the remaining epoch, a learning rate was 10 times lower. In the case of the backbone network, a learning rate of 5×10^{-5}, which is 10 times lower, was applied. We trained our BSA model for a total of 150 epochs with a batch size of 16.

In the case of the SBD dataset, since there is no seed data, they are generated through sampling from the GT masks. We sampled training seed data through 3 strategies as in [1]. Furthermore, we adopted the refinement and Zoom-In skill of f-BRS-B [6] in our inference step to improve performance. The f-BRS-B algorithm, like us, has the purpose of preserving seed information, but unlike us, it is applied to the decoder stage and is also a refinement technique so that it can be orthogonal to our algorithm.

IV. Experiments

Datasets. For evaluation, we used four standard segmentation benchmark datasets: SBD [7], GrabCut [8], Berkeley [9], and DAVIS [10] datasets. We experimented with the validation set of the SBD dataset. It consists of a total of 2,820 images with a total number of 6,671 instance object masks. The GrabCut dataset consists of a total of 50 images, and each image has segmentation information for one object. The test image of the Berkeley

TABLE I
COMPARISON WITH OTHER INTERACTIVE SEGMENTATION METHODS (NOC 85% AND NOC 90%). THE BEST RESULTS ARE SHOWN IN BOLD, AND THE SECOND-BEST RESULTS ARE SHOWN IN ITALICS

Method	Train Set	SBD		GrabCut		Berkeley	DAVIS	
		@85	*@90*	*@85*	*@90*	*@90*	*@85*	*@90*
GC [11]	-	13.60	15.96	7.98	10.00	14.22	15.13	17.41
RW [12]	-	12.22	15.04	11.36	13.77	14.02	16.71	18.31
GSC [14]	-	12.69	15.31	7.10	9.12	12.57	15.35	17.52
DIOS [1]	*SBD*	9.22	12.80	5.08	6.08	-	9.03	12.58
RISNet [3]	*VOC*	-	-	-	5.00	6.03	-	-
LD [2]	*SBD*	7.41	10.78	3.20	4.79	-	5.05	9.57
ITIS [33]	*VOC*	-	-	-	5.60	-	-	-
FCTSFN [16]	*VOC*	-	-	-	3.76	6.49	-	-
CMG [4]	*VOC*	-	-	-	3.58	5.60	-	-
BRS [5]	*SBD*	6.59	9.78	2.60	3.60	5.08	5.58	8.24
DIOS [17]	*VOC*	-	-	-	**1.96**	*4.31*	-	-
f-BRS-B [6]	*SBD*	4.81	7.73	2.30	2.72	4.57	*5.04*	7.41
Baseline-*res101*	*SBD*	5.25	8.31	3.12	3.86	5.27	5.15	7.59
BSA	*SBD*	*4.63*	*7.44*	*2.22*	2.58	4.97	**5.01**	*7.17*
BSA + f-BRS-B	*SBD*	**4.44**	**7.30**	**2.00**	*2.42*	**4.23**	5.20	**7.06**

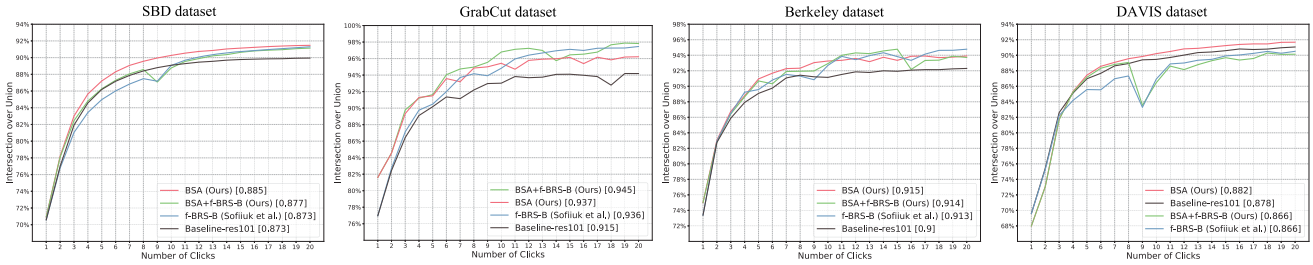

Fig. 4. Click-IoU curve graph for each dataset. The number in the legend indicates the AuC score.

dataset is 96 and has 100 object masks, which means some images contain information about multiple segments. Finally, the DAVIS dataset is a dataset for video object segmentation and consists of 50 types of video. We sampled 10% of the frames as in [5] and used them for evaluation.

Metrics. The most widely used metric for segmentation is the intersection over union (IoU). In this work, we evaluate the performance of interactive segmentation algorithms using the Number of Clicks (NoC) based on IoU like in [1], [6]. NoC is a method of counting the number of inputs required to reach the target IoU when a robot user [14] is applied to an interactive segmentation system. As a method to simulate an actual annotation environment, it is a metric suitable for evaluating performance that minimizes user effort. We measured NoC for two IoU thresholds of 85% and 90%. Also, we used the value of the area under curve (AuC) in the Click-IoU graph as another metric. For the AuC value, we used a normalized value for easy identification.

We have compared our algorithm with most existing state-of-the-art methods including the classic approaches like GC [11], RW [12], GSC [14], and recent deep learning based techniques such as DIOS [1], RIS-Net [3], LD [2], ITIS [33], FCTSFN [16], CMG [4], BRS [5], DIOS [17], and f-BRS-B [6]. They can also be divided into three categories: Methods without training, methods trained with the PASCAL VOC [34] dataset, and methods trained with the SBD dataset. Therefore, it is not a completely fair comparison, but we can compare using a common dataset. Table I shows the experimental results. The proposed algorithm, BSA, recorded the least number of clicks in most results. For GrabCut datasets, DIOS [17] gives the best results, but BSA shows the best among algorithms that are trained on the SBD dataset. Meanwhile, the baseline (backbone) network we used is the same as the baseline of f-BRS-B [6]. Even when we apply only the BSA module to the baseline, it shows better performance than f-BRS-B. Furthermore, when f-BRS-B refinement step is additionally applied, the performance gain becomes much larger.

Fig. 4 shows a graph of changes in IoU according to click. We compared our algorithm with [6], which shows state-of-the-art performance. In the case of f-BRS-B, the refinement is applied to our baseline so that the value may be slightly different from their original paper. In all datasets, including SBD, our BSA or BSA+f-BRS-B algorithm records the best AuC. Unlike the results in Table I, in some datasets in Fig. 4, BSA shows better results than BSA+f-BRS-B. Due to the characteristics of f-BRS-B showing worse results than baseline in AuC metric, it shows similar pattern when combined with BSA.

V. CONCLUSION

In this paper, we proposed novel bi-directional seed attention (BSA) network for interactive segmentation. By adding a simple BSA module to the backbone segmentation network, we enhanced the semantic information around the seed to obtain a better segmentation mask. We demonstrated the superiority of the proposed network over existing state-of-the-art methods on various benchmark datasets.

REFERENCES

[1] N. Xu, B. Price, S. Cohen, J. Yang, and T. S. Huang, "Deep interactive object selection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 373–381.

[2] Z. Li, Q. Chen, and V. Koltun, "Interactive image segmentation with latent diversity," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 577–585.

[3] J. H. Liew, Y. Wei, W. Xiong, S.-H. Ong, and J. Feng, "Regional interactive image segmentation networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2746–2754.

[4] S. Majumder and A. Yao, "Content-aware multi-level guidance for interactive instance segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 11594–11603.

[5] W.-D. Jang and C.-S. Kim, "Interactive image segmentation via backpropagating refinement scheme," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5292–5301.

[6] K. Sofiiuk, I. Petrov, O. Barinova, and A. Konushin, "F-BRS: Rethinking backpropagating refinement for interactive segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8623–8632.

[7] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, and J. Malik, "Semantic contours from inverse detectors," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 991–998.

[8] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, 2004.

[9] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2001, pp. 416–423.

[10] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 724–732.

[11] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in ND images," in *Proc. 8th IEEE Int. Conf. Comput. Vis.*, 2001, pp. 105–112.

[12] L. Grady, "Random walks for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1768–1783, Nov. 2006.

[13] T. H. Kim, K. M. Lee, and S. U. Lee, "Generative image segmentation using random walks with restart," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 264–275.

[14] V. Gulshan, C. Rother, A. Criminisi, A. Blake, and A. Zisserman, "Geodesic star convexity for interactive image segmentation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 3129–3136.

[15] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.

[16] Y. Hu, A. Soltoggio, R. Lock, and S. Carter, "A fully convolutional two-stream fusion network for interactive image segmentation," *Neural Netw.*, vol. 109, pp. 31–42, 2019.

[17] J. H. Liew, S. Cohen, B. Price, L. Mai, S.-H. Ong, and J. Feng, "Multiseg: Semantically meaningful, scale-diverse segmentations from minimal user input," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 662–670.

[18] F. Wang *et al.*, "Residual attention network for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6450–6458.

[19] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7132–7141.

[20] M. Choi, H. Kim, B. Han, N. Xu, and K. M. Lee, "Channel attention is all you need for video frame interpolation," in *Proc. 34th AAAI Conf. Artif. Intell.*, 2020, pp. 10663–10671.

[21] Z. Zhu, W. Wu, W. Zou, and J. Yan, "End-to-end flow correlation tracking with spatial-temporal attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 548–557.

[22] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 7354–7363.

[23] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille, "Attention to scale: Scale-aware semantic image segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3640–3649.

[24] J. Fu *et al.*, "Dual attention network for scene segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3141–3149.

[25] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-cross attention for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 603–612.

[26] J. Park, S. Woo, J.-Y. Lee, and I. S. Kweon, "BAM: Bottleneck attention module," in *Proc. Brit. Mach. Vis. Conf.*, 2018, p. 147.

[27] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.

[28] H. Li, G. Chen, G. Li, and Y. Yu, "Motion guided attention for video salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7273–7282.

[29] Q. Cui, H. Sun, Y. Li, and Y. Kong, "A deep bi-directional attention network for human motion recovery," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, 2019, pp. 701–707.

[30] M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi, "Bidirectional attention flow for machine comprehension," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–13.

[31] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 833–851.

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[33] S. Mahadevan, P. Voigtlaender, and B. Leibe, "Iteratively trained interactive segmentation," in *Proc. Brit.Mach. Vis. Conf.*, 2018, p. 212.

[34] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, pp. 303–338, 2010.