ACDNet with ASPP for Camouflaged Object Detection

Qihui Zhu
Faculty of Urban Construction BJUT, Beijing University of Technology, Beijing, China
18398090@bjtu.edu.cn

Abstract. This paper proposes a camouflage identification network ACDNet, which can perform end-to-end search and identification simply and efficiently. Use ASPP (Atrous Spatial Pyramid Pooling) to enhance the image features of camouflaged object, and then connect the Parallel Decoder (PD) module to get the global map, and then use the Group-Reversal Block (GRB) module to connect sequentially, refine the details, and finally perform deep supervision learning got result. Compared with the current 13 methods, ACDNet has a significant performance improvement in accuracy and similarity, and it is one of the potential solutions for camouflage identification.

Keywords: Camouflaged object detection; ACDNet; ASPP; PD; GRB.

1. Introduction

In nature, a large number of animals use camouflage to improve survival chances. A school of fish moving through coral can change to a background-like color to hide from predators. An octopus can adjust its color at will to camouflage itself. It is the result of natural selection as well as a self-protection mechanism. The camouflaged object detection (COD) task aims at searching and segmenting camouflaged objects that blend in with their RGB and depth image surroundings. Discovering camouflaged objects is challenging for both humans and artificial intelligence. It is important for rescue team, agricultural pest detection, medical imaging and military. So far, researchers have explored many camouflaged object detection (COD) strategies and have achieved inspiring results. However, COD still has the following challenges currently:

1. How to improve the localization of camouflaged object and accurately segment the edges: Most of the COD networks studied so far are only able to locate the rough location of camouflage objects. However, how to refine the detail information to ensure the accuracy of the edges is still challenging. Previously, most COD networks did not take low-level features into consideration. Although they tend to introduce noises, they can provide a wealth of detailed information. How to introduce low-level features and make positive impact on detail refinement has become a major challenge.

2. Lack of large-scale datasets and how to improve the speed of camouflage detection: The largest available camouflage dataset is the COD10K proposed by Fan in article [7], which contains 10,000 images of camouflaged object. However, more complex and challenging datasets such as artificial camouflage and large-scale camouflaged object have not been systematically assembled. In addition, one of the current challenges is to simplify the training neural networks with large amounts of data to increase its speed.
Targeting these drawbacks, this study presents a ASPP Camouflage Detection Network (ACDNet). The ASPP module is used to expand the receptive field, followed by the connection of a parallel decoder (PD) and a group-reversal block (GRB) for deep learning and ultimately effective detection of camouflaged object in the pictures. The contributions of this work are summarized as follows:

1. Expanding the receptive field could better exploit the differences between foreground and background, which will have a positive impact on advanced image feature processing. But this will result in a loss of local information at the same time. To solve this problem, ACDNet introduces an ASPP module based on null convolution, which both keeps the resolution from decreasing too much and expands the receptive field.

2. In order to make full use of the features from different levels, the present study proposes a parallel decoder to combine the high-level features effectively. Thus, the approximate location of the camouflaged object could be located accurately. In addition, to fully obtain useful information from the depth image and to effectively improve the details, a back-propagation module refining the edges layer by layer for multiple times is introduced. The ability to fuse features at all levels could be improved.

3. Applied in 2 current datasets and evaluated with 4 evaluation methods, ACDNet shows excellent performance in accuracy and similarity compared with 13 methods, which indicates that this network structure may be a potential solution for camouflaged object detection task.

2. Related Work

2.1. Camouflaged Object Detection

2.1.1. Definition. Object detection is currently a popular area of computer vision research, and camouflaged object detection is one of the most difficult parts. Camouflaged objects are more similar to the background than salient objects, making their location and details more challenging to identify.

2.1.2. Present Study and Future Research. Numerous COD researches have been conducted. Trung-Nghia Le proposed the Anabranch Network in article [3]. They built a new dataset of camouflaged object and proposed an end-to-end network with a second branch for predicting the probability of containing artifacts in an image and fusing it into the main branch for segmentation to improve the accuracy. In article [4], QIN proposed a boundary-aware salient target detection network (BASNet), which consists of a Predict Module to obtain a coarse salient image and an RRM module to overcome the "coarse". A new hybrid loss is also proposed, fusing BCE, SSIM and IoU losses for pixel-level, patch-level and map-level respectively. Yunqiu Lv proposed the Joint Localization and Segmentation Framework in his article [6], where the Fixation Decoder generates the discriminative region. This region has a higher contrast with the surrounding environment, where the camouflaged object approximately locates. Camouflage Decoder adopts the idea of reverse attention, obtains structured information to generate the final prediction map. The SINet proposed by Fan in article [2] is also one of the references of present research. It consists of two main modules, one is the search module (SM) and the other is the identification module (IM). The former is responsible for searching the camouflaged objects and the latter is used for accurate object detection.

2.2. Receptive Field and Its Application

In convolutional neural network, the definition of Receptive Field is the size of the area mapped on the input image by the pixel points on the feature map outputted from each layer of the convolutional neural network. Receptive field, which is generally applied to salient target recognition, cannot meet the needs of more complex camouflaged objects detection (COD) tasks. Currently, there are several ways to enhance the perceptual field as follows.
2.2.1. **Spatial Pyramid Pooling (SPP).** He KaiMing proposed an optimization module called SPP in article [1]. The SPP layer divides the feature map into grids of different sizes and does maximum pooling within each grid so that the subsequent fully connected layers could get a fixed output.

2.2.2. **Receptive Field Block (RFB).** The RFB module was proposed in the paper (ECCV2018: Receptive Field Block Net for Accurate and Fast Object Detection). It aims to simulate the receptive field of human vision to enhance the feature extraction capability of the network. The structure of RFB is based on the idea of Inception. Through adding the null convolution to Inception, the perceptual field could be effectively enhanced.

2.2.3. **Pyramid Pooling Module (PPM).** Generally speaking, the deeper the network is, the larger the receptive field is. However, there is a gap between the theoretical receptive field and the actual one in the network (the actual receptive field is smaller than the theoretical one), which makes the network unable to effectively fuse the global feature information. PPM divides the feature map extracted from the previous network into two branches. One branch uses Global Average Pooling (GAP) in multiple regions, and finally the two branches are fused to solve the problem mentioned above.

2.2.4. **Atrous Spatial Pyramid Pooling (ASPP).** The ASPP module is based on SPP, where the receptive field is expanded by using the convolution of cavities with different resolutions. This not only improves the accuracy but also is faster and more effective. ASPP is adopted in present research to expand the receptive field. More details will be elaborated in later sections.

2.3. **Attention Mechanism**

Based on the strong capability of both feedforward and recurrent networks, the advantage of the attention mechanism is to solve the limitation of computational power and optimization algorithm. The former is that the neural network has to remember too much information but this is restricted by its computational power. The latter is that although optimization operations such as local connectivity, weight sharing, and pooling can make neural networks simpler, alleviate the contradiction between model complexity and expressiveness effectively, its "memory" capability is not powerful enough for problems like recurring neural networks over long distances. Thus, attention mechanism was created and mainly has following two categories.

1. **Focus:** It is a top-down conscious attention. This active attention is a task-dependent attention which actively and consciously focused on an object with predetermined purpose.

2. **Saliency-Based Attention:** It is a bottom-up conscious attention. This passive attention is a task-independent salience-based attention driven by external stimuli, which does not require active intervention. Max-pooling and gating mechanisms can be categorized as bottom-up salience-based attention mechanisms.

The attention mechanism can be executed in three steps: (1) inputting information; (2) calculating attention distribution; (3) calculating the average weight of the input information based on the attention distribution.

Present study refers to the RGB-D salient object detection network proposed in the article [7]. The network first constructs a lightweight depth stream by learning from scratch instead of using the ImageNet pre-trained backbone network, which can extract complementary features more efficiently and reduce redundancy. Then, RGB and depth features are alternately fed into the proposed bootstrap residual blocks to reduce their mutual degradation. By assigning progressive bootstrapping in the stacked GR blocks in the output of each side, false detections and missed parts can be well corrected. Superiority is also showed in efficiency and model size.

3. **Method**

The high similarity between the camouflaged object and the background makes artifact recognition more challenging compared to salient one. Atrous Spatial Pyramid Pooling (ASPP) is adopted in present
research to enhance the camouflaged object features. Subsequently, the parallel decoder module (PD) is connected to obtain the global mapping map, and then the group-reversal block (GRB) is used to sequentially connect and refine the detailed information to perform deep supervised learning. The overall ACDNet is shown in Figure 1.

Figure 1. ACDNet.

3.1. ASPP

In conventional neural networks for semantic segmentation of images or videos, downsampling is often used to expand the receptive field and abstract the feature information. This works well for advanced image processing tasks, but not for predictive image tasks with input resolutions such as output. There are two main problems. First, information is lost during downsampling. The resolution of the image drops dramatically in the process of continuous abstraction, resulting in the loss of local feature information and detailed information of the image. Although linear interpolation upsampling can compensate for this, it is not effective. In addition, the size of the input image is not fixed. A common processing method is image pyramiding, which means that the original image is stretched, scaled to different sizes and then input to the same network before fusion. This approach can improve the accuracy but at a very low rate. Therefore, we need to find out a method that can expand the receptive field, compensate for the information loss caused by downsampling, and increase the speed at the same time.

The Atrous spatial pyramid pooling (ASPP) module is inspired by the SPP module. It uses multiple parallel dilated convolution layers with different sampling rates for the purpose of expanding the receptive field. ASPP captures multi-scale information by convolving cavities with different expansion rates. This not only improves the accuracy but also is faster.

\[ p = 2(dilate rate - 1) * (k - 1) + k \]

Where \( p \) is the null convolution field size, \( k \) the convolution kernel size, and \( dilate rate \) is the dilated convolution rate.

In this model, one convolution and three dilated convolutions with \( rate = \{6, 12, 18\} \) of 256 output channels are used, including a batch normalization (BN) layer. As showing in Figure 2, the global average pooling of the image-level features is convolved with the result of the dilated convolution and then fused to obtain the final result.
3.2. Parallel decoder (PD)

Currently, existing salient object detection and camouflaged object detection usually classify features into two types: high-level features and low-level features. High-level features are more suitable for localizing camouflaged object as a whole. They contain rich global environmental information that facilitates camouflaged object localization. Low-level features require greater spatial resolution, consume more computational resources, and contribute less to performance. However, they carry a large amount of detailed information, which can effectively improve the uncertain edges in COD tasks. When an image $I$ of size $h \times w$ is input to this network, the multi-scale features are divided into two parts: high-level features and low-level features. $\{ch_i, i = 1, 2\}$ stands for low-level features and $\{ch_i, i = 3, 4, 5\}$ stands for the high-level features, where the resolution of each feature is $\left\lfloor \frac{h}{2^{a_1}}, \frac{w}{2^{a_2}} \right\rfloor$.

Present research makes full use of the high-level multimodal features to mine semantic information and to suppress background interference in a parallel combination. The high-level features are enhanced by the ASPP and results are defined as: $\{ch_i^a, i = 3, 4, 5; a = 1\}$ Then, they enter the partial decoder (PD).

\[
\begin{align*}
U1 &= \text{Conv3}(\text{Up}(ch_i^1)) * ch_i^1 \\
U2 &= \text{Conv3}(\text{Up}(U1)) * \text{Conv3}(\text{Up}(ch_i^1)) * ch_i^1 \\
c1 &= \text{Conv3}(\text{Up}[\text{Conv3}(\text{C}(\text{Conv3}(\text{Up}(ch_i^1)), U1))]) \\
c2 &= \text{C}(U2, c1)
\end{align*}
\]

Where $\text{Conv3}$ represents a $3 \times 3$ convolution as well as the sequential operation of a BN layer; $\text{Conv1}$ represents a $1 \times 1$ convolution as well as the sequential operation of a BN layer. $\text{Up}$ represents the upsampling operation. Finally, these different features are combine by a concatenation operation $\text{C}$. The whole partial decoder process is shown in Fig 3.
Finally, after two convolutions and one convolution, the global mapping map $P_o$ is obtained.

### 3.3. Group-Reversal Block (GRB)

In camouflaged object detection, objects’ locations are first roughly located and then these regions are accurately marked by examining the edges. Global mapping $P_o$ with unstructured details have been generated previously to provide the coarse locations of camouflaged objects. Subsequently, a group-reversal block (GRB) is proposed which can mine detail information progressively. This system sequentially mines the complementary region details by erasing the existing estimated regions from the high-level side output features. Results will be applied to the next group-reversal block operation to obtain the final prediction map $P_f$. The specific process is shown in the figure 1.

**Figure 3.** Partial Decoder (PD) process

First, the reverse guiding principle is carried out for $\{P_i, i = 3, 4, 5\}$. In other words, the reverse bootstrap is obtained through the Sigmoid function:

$$
E(\sigma(\delta(P_i)))
$$

$E$ stands for the inverse operator that subtracts the input from the all-1 matrix, $\sigma$ represents the Sigmoid function. Here $\delta$ is a $\times 2$ upsampling operation when $i = 3, 4$ and a $\times 4$ downsampling operation when $i = 5$. After that, $A_i^1$ and $ch_i^1$ are input into the GR module together. $a$ is the number of operation times. This is the initial state before inputting into the GR module, which means $a = 1$.

Subsequently, the ASPP-enhanced features $\{ch_i^a, i = 3, 4, 5\}$ are divided into $m$ dimensions along the direction of channel, and then they are inserted into $A_i^a$ respectively. As shown in the Figure

**Figure 4.** Specific process of GR

First, the reverse guiding principle is carried out for $\{P_i, i = 3, 4, 5\}$. In other words, the reverse bootstrap is obtained through the Sigmoid function:

$$
E(\sigma(\delta(P_i)))
$$

$E$ stands for the inverse operator that subtracts the input from the all-1 matrix, $\sigma$ represents the Sigmoid function. Here $\delta$ is a $\times 2$ upsampling operation when $i = 3, 4$ and a $\times 4$ downsampling operation when $i = 5$. After that, $A_i^1$ and $ch_i^1$ are input into the GR module together. $a$ is the number of operation times. This is the initial state before inputting into the GR module, which means $a = 1$.

Subsequently, the ASPP-enhanced features $\{ch_i^a, i = 3, 4, 5\}$ are divided into $m$ dimensions along the direction of channel, and then they are inserted into $A_i^a$ respectively. As shown in the Figure
3.4. **Loss function**

According to article [2], loss function is defined as

$$L = L_{IoU} + \omega L_{BCE}$$

$L_{IoU}$ is a globally constrained and locally weighted IoU loss that increases the weights of difficult pixel sample points to highlight their importance. $\omega L_{BCE}$ is a globally constrained and locally weighted BCE loss that focuses more on difficult pixel sample points rather than assigning the same weight to all pixels. The validity of both has been verified in related fields. Meanwhile, $P_3, P_4, P_5, P_6$ are deeply supervised. The function of total loss is $L_f = L(P_6, G) + \sum_{i=3}^{5} L(P_i, G)$.

4. **Evaluation Standard**

The detection of camouflaged objects is more complex compared to that of salient. The present study uses four evaluation metrics to evaluate accuracy and similarity comprehensively.

4.1. **S-measure ($S_a$)**

This metric can evaluate structural similarity. The objects in camouflaged object are usually complex, so validation of the S-rater is necessary. It is defined as.

$$S_a = (1 - \alpha) S_o(P_f, G) + \alpha S_r(P_f, G)$$

Where $\alpha$ is used to control the balance coefficient between object similarity level $S_o$ and region similarity $S_r$. The default setting is $\alpha = 0.5$, $G$ is the true value, $P_f$ is the final prediction.

4.2. **Mean absolute error (MAE)**

Mean absolute error (MAE) has been widely used in salient object detection evaluation. $MAE(M)$ is used to evaluate the pixel-level accuracy between the predicted image and the true image. Its disadvantage is that it cannot evaluate where errors occur. It is defined as

$$MAE = \frac{1}{w \times h} \sum_{x}^{w} \sum_{y}^{h} |P_f(x, y) - G(x, y)|$$

Where $w$ and $h$ stand for the width and height of the true image $G$ respectively, and $(x, y)$ is the coordinates of each pixel.

4.3. **E-measure ($E_\phi$)**

To compensate for these shortcomings, Fan et al. proposed the E-measure ($E_\phi$). It considers both pixel-level information matching and image-level information statistics. It is also suitable for evaluating the overall and local accuracy of artefact detection results. It is defined as

$$E_\phi = \frac{1}{w \times h} \sum_{x}^{w} \sum_{y}^{h} \phi(P_f(x, y) - G(x, y))$$
Where $\phi$ is an enhanced alignment matrix.

### 4.4. F-measure ($F^\alpha_\beta$)

In addition, recent research has shown that the weighted F-measure ($F^\alpha_\beta$) is more reliable than traditional evaluation methods. Therefore, this evaluation method is also adopted in current research.

$$F^\alpha_\beta = \frac{2PR}{P + R}$$

Where $P$ stands for precision and $R$ stands for recall. When $F^\alpha_\beta$ is high, then it can indicate that the test method is more effective.

### 5. Results and Data Analysis

#### 5.1. Dataset

Present research used CHAMELEON, CAMO and COD10K for tests and training. The CHAMELEON dataset was built in 2018 with only 76 images and manually labeled with ground truths (GTs). The CAMO dataset was built in 2019, covering 2.5K pictures and 8 categories. The COD10K dataset was assembled by Fan in 2020. It contains 10K images covering 78 camouflaged object categories, such as aquatic, amphibians, and terrestrial, etc. The high-quality annotations in this dataset can provide deeper insight into the performance of algorithms.

#### 5.2. Implementation Details

ACDNet is implemented in PyTorch and trained with the Adam optimizer. During the training stage, the batch size is set to 24, and the learning rate starts at 1e-4. The whole training time is about 5 hours by end-to-end strategy. The running time is measured on the Windows10 system 2080Ti $\times$ 6 and RTX. Each image is adjusted to 352×352.

#### 5.3. Quantitative Analysis

Two experiments are conducted to validate the learning ability of the model on CAMO and COD10K datasets. Table 1 shows the test results.

| Baseline Models | CAMO-Test | COD10K-Test |
|-----------------|-----------|-------------|
|                 | $S^a_\phi$ | $E^a_\phi$ | $F^\alpha_\beta$ | $M$ | $S^a_\phi$ | $E^a_\phi$ | $F^\alpha_\beta$ | $M$ |
| 2017 FPN        | 0.684     | 0.677      | 0.483           | 0.131| 0.697      | 0.691      | 0.411           | 0.075|
| 2017 MaskRCNN   | 0.574     | 0.715      | 0.43            | 0.151| 0.613      | 0.748      | 0.402           | 0.08 |
| 2017 PSPNet     | 0.663     | 0.659      | 0.455           | 0.139| 0.678      | 0.68       | 0.377           | 0.08 |
| 2018 UNet++     | 0.599     | 0.653      | 0.392           | 0.149| 0.623      | 0.672      | 0.35            | 0.086|
| 2018 PiCANet    | 0.609     | 0.584      | 0.356           | 0.156| 0.649      | 0.643      | 0.322           | 0.09 |
| 2019 MSRRCNN    | 0.617     | 0.669      | 0.454           | 0.133| 0.641      | 0.706      | 0.419           | 0.073|
| 2019 BASNet     | 0.618     | 0.661      | 0.413           | 0.159| 0.634      | 0.678      | 0.365           | 0.105|
| 2019 PFANet     | 0.659     | 0.622      | 0.391           | 0.172| 0.636      | 0.618      | 0.286           | 0.128|
| 2019 CPD        | 0.726     | 0.729      | 0.55            | 0.115| 0.747      | 0.77       | 0.508           | 0.059|
| 2019 HTC        | 0.476     | 0.442      | 0.174           | 0.172| 0.548      | 0.52       | 0.221           | 0.088|
| 2019 EGGNet     | 0.732     | 0.768      | 0.583           | 0.104| 0.737      | 0.779      | 0.509           | 0.056|
| 2019 ANet-SRM   | 0.682     | 0.685      | 0.484           | 0.126| †          | †          | †              | †   |
| SINet’20 Training | 0.751      | 0.771      | 0.606           | 0.1  | 0.771      | 0.806      | 0.551           | 0.051|
| ACDNet          | 0.820     | 0.873      | 0.744           | 0.071| 0.815      | 0.883      | 0.689           | 0.041|

8
Results show that ACDNet performs better in four evaluation metrics of CAMO and COD10K tests. Due to the ASPP expands the receptive field, parallel cascade module and GRB inverse module, it outperforms the cutting-edge SINet and its robustness is proved.

Comparing ACDNet with SINet in the CAMO-Test dataset, ACDNet exceeds SINet’s $S_\alpha$ by 0.069%, $E_\phi$ by 0.102%, $F_\omega^m$ by 0.138% and $M$ by 2.9%. Compared to 2017 MaskRCNN, ACDNet exceeds 0.246%, 0.158%, 0.314% and 0.08% in $S_\alpha$, $E_\phi$, $F_\omega^m$ and $M$ respectively. Compared to the 2019 CPD, ACDNet exceeds 0.094%, 0.144%, 0.194% and 0.044% in $S_\alpha$, $E_\phi$, $F_\omega^m$ and $M$ respectively.

In the COD10K dataset test, ACDNet exceeds SINet by 0.044% for $S_\alpha$, 0.077% for $E_\phi$, 0.138% for $F_\omega^m$ and 0.01% for $M$ reduction. Compared to classical 2017 MaskRCNN, ACDNet exceeds 0.202 % for $S_\alpha$, 0.135% for $E_\phi$, 0.287 % for $F_\omega^m$ and its $M$ decreases by 0.039 %. Compared to 2019 CPD, ACDNet exceeds 0.27 % for $S_\alpha$, 0.135% for $E_\phi$, 0.287 % for $F_\omega^m$ and its $M$ decreases by 0.039 %.

In all, ACDNet performs better in all evaluation metrics for both test sets. It indicates that ACDNet has better performance both in accuracy and similarity. Therefore, ACDNet could be one of the potential solutions to the camouflaged object detection task.

5.4. Qualitative Analysis

Figure 5. provides the performance of various camouflaged object detection network. Here, three representative images with identification difficulties are randomly selected. The first picture is partially obscured, the second one has difficulty to determine the edge, and the third one has smaller, more detailed and informative object. It can be found that in figure (a), SINet misidentifies the camouflage object widely, PraNet does identify the camouflage information of the fish’s tail, and ACDNet exceeds the performance of the above two overall in spite of small inaccuracy in detailed part below. In Fig. (b), the three methods perform well overall, but ACDNet is more accurate in refining the information of the fish’s tail. As for figure (c), the object is smaller and it is harder to refine the detailed information. In this case, the artifact localization of SINet is very ambiguous, and ACDNet refines the legs of the crab more compared to PraNet. Therefore, the qualitative analysis shows that ACDNet can locate the artifacts and display detailed information more accurately. This reflects the robustness of the framework in this article.

![Figure 5. Qualitative analysis](image-url)
5.5. Ablation experiment

Table 2. CAMO-Test and COD10K-Test results

| Baseline Models           | CAMO-Test       | COD10K-Test       |
|--------------------------|-----------------|-------------------|
|                          | $S_\alpha$      | $E_\phi$         | $F_\phi$ | $M$         | $S_\alpha$      | $E_\phi$         | $F_\phi$ | $M$         |
| Backbone                 | 0.798           | 0.863            | 0.696    | 0.080       | 0.789           | 0.836            | 0.628    | 0.048       |
| GRB+ Backbone            | 0.807           | 0.876            | 0.725    | 0.074       | 0.813           | 0.870            | 0.686    | 0.041       |
| ASPP+PD+ Backbone        | 0.812           | 0.878            | 0.732    | 0.073       | 0.814           | 0.873            | 0.690    | 0.040       |
| ACDNet                   | **0.820**       | **0.873**        | **0.744**| **0.071**   | **0.815**       | **0.883**        | **0.689**| **0.041**   |

In this section, the validity of ASPP module, PD module and GRB is verified. First, we further explored the contribution of GRB, and the results are shown in the second row of Table 2. It can be seen that various indicators in the two data sets have greatly improved compared to backbone. Subsequently, the ablation experiment of the ASPP and PD modules was performed. The results are shown in the third row of Table 2. It can be seen that the introduction of various modules can make the camouflage identification network ACDNet more accurate.

6. Conclusion

This paper proposes a camouflage identification network ACDNet, which can perform end-to-end search and identification simply and efficiently. Compared with the current 13 methods, ACDNet has a significant performance improvement in accuracy and similarity. But there is still the problem of accurate edge segmentation. In summary, ACDNet is one of the potential solutions for camouflage identification.

References

[1] He Kaiming, Zhang Xiangyu, Ren Shaoqing, Sun Jian. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. IEEE transactions on pattern analysis and machine intelligence, 2015, 37(9).
[2] Deng-Ping Fan, Ge-Peng Ji, Guolei Sun, Ming-Ming Cheng, Jianbing Shen, Ling Shao. Camouflaged Object Detection. CVPR 2020.
[3] Trung-Nghia Le, Tam V. Nguyen, Zhongliang Nie, Minh-Triet Tran, Akihiro Sugimoto. Anabranch network for camouflaged object segmentation. CVIU, 2019.
[4] Xuebin Qin, Deng-Ping Fan*, Chenyang Huang, Cyril Diagne, Zichen Zhang, Adrià Cabeza Sant’Anna, Albert Suàrez, Martin Jagersand, Ling Shao. Boundary-Aware Segmentation Network for Mobile and Web Applications. arXiv, 2021.
[5] Qiang Zhai, Xin Li, Fan Yang*, Chenglizhao Chen, Hong Cheng, Deng-Ping Fan, Mutual Graph Learning for Camouflaged Object Detection, CVPR, 2021.
[6] Yunqiu Lv, Jing Zhang, Yuchao Dai*, Li Aixuan, Bowen Liu, Nick Barnes, Deng-Ping Fan, Simultaneously Localize, Segment and Rank the Camouflaged Objects, CVPR, 2021.
[7] S.Chen and Y.Fu, progressively guided alternate refinement network for rgb-d salient object detection, in Eur. Conf. Comput.Vis. Springer, 2020, pp. 520-538