Camouflaged Object Detection

Deng-Ping Fan[1,2] Ge-Peng Ji[3] Guolei Sun[4] Ming-Ming Cheng[2] Jianbing Shen[1] * Ling Shao[1]
[1] Inception Institute of Artificial Intelligence, UAE [2] College of CS, Nankai University, China
[3] School of Computer Science, Wuhan University, China [4] ETH Zurich, Switzerland

http://dpfan.net/Camouflage/

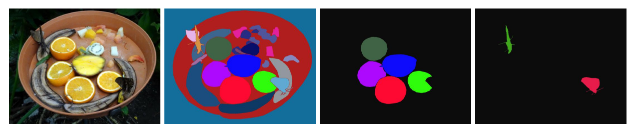
Figure 1: Examples from our *COD10K* dataset. Camouflaged objects are concealed in these images. Can you find them? **Best viewed in color and zoomed-in**. Answers are presented in the supplementary material.

Abstract

*We present a comprehensive study on a new task named camouflaged object detection (COD), which aims to identify objects that are "seamlessly" embedded in their surroundings. The high intrinsic similarities between the target object and the background make COD far more challenging than the traditional object detection task. To address this issue, we elaborately collect a novel dataset, called **COD10K**, which comprises 10,000 images covering camouflaged objects in various natural scenes, over 78 object categories. All the images are densely annotated with category, bounding-box, object-/instance-level, and matting-level labels. This dataset could serve as a catalyst for progressing many vision tasks,* e.g., *localization, segmentation, and alpha-matting,* etc. *In addition, we develop a simple but effective framework for COD, termed Search Identification Network (**SINet**). Without any bells and whistles, SINet outperforms various state-of-the-art object detection baselines on all datasets tested, making it a robust, general framework that can help facilitate future research in COD. Finally, we conduct a large-scale COD study, evaluating 13 cutting-edge models, providing some interesting findings, and showing several potential applications. Our research offers the community an opportunity to explore more in this new field. The code will be available at: https://github.com/DengPingFan/SINet/*

1. Introduction

Can you find the concealed object(s) in each image of Fig. 1? Biologists call this *background matching camouflage* [9], where an animal attempts to adapt their body's coloring to match "perfectly" with the surroundings in order to avoid recognition [48]. Sensory ecologists [57] have found that this camouflage strategy works by deceiving the visual perceptual system of the observer. Thus, addressing *camouflaged object detection* (**COD**) requires a significant amount of visual perception [60] knowledge. As shown in Fig. 2, the high intrinsic similarities between the target object and the background make COD far more challenging than the traditional salient object detection [1, 5, 17, 25, 62–66, 68] or generic object detection [4, 79].



(a) Image (b) Generic object (c) Salient object (d) Camouflaged object

Figure 2: Given an input image (a), we present the ground-truth for (b) panoptic segmentation [30] (which detects **generic** objects [39, 44] including stuff and things), (c) **salient** instance/object detection [16, 33, 61, 76] (which detects objects that grasp human attention), and (d) the proposed **camouflaged** object detection task, where the goal is to detect objects that have a similar pattern (*e.g.*, *edge*, *texture*, or *color*) to the natural habitat. In this case, the boundaries of the two butterflies are blended with the bananas, making them difficult to identify.

In addition to its scientific value, COD is also beneficial for applications in the fields of computer vision (for search-and-rescue work, or rare species discovery), medical image segmentation (*e.g.*, polyp segmentation [14], lung infection segmentation [18, 67]), agriculture (*e.g.*, locust detection to prevent invasion), and art (*e.g.*, for photo-realistic blending [21], or recreational art [6]).

Currently, camouflaged object detection is not well-

* Corresponding author: Jianbing Shen (shenjianbingcg@gmail.com).

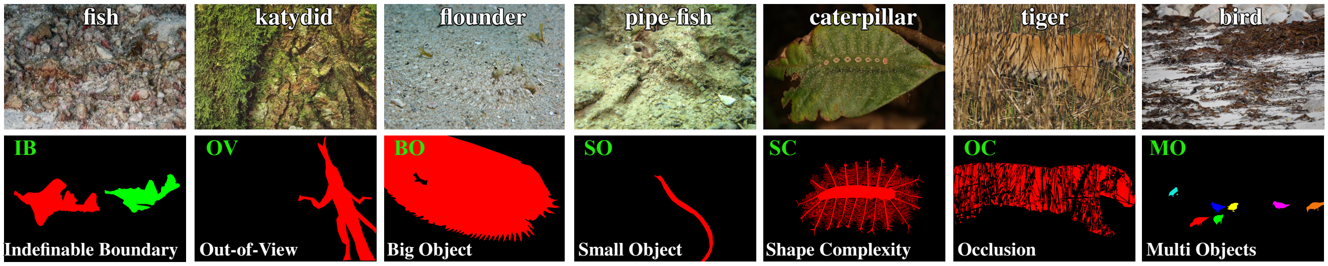


Figure 3: Various examples of challenging attributes from our *COD10K*. See Tab. 2 for details. Best viewed in color, zoomed in.

studied due to the lack of a sufficiently large dataset. To enable a comprehensive study on this topic, we provide two contributions. First, we carefully assembled the novel ***COD10K*** dataset exclusively designed for COD. It differs from current datasets in the following aspects:

- It contains 10K images covering 78 camouflaged object categories, such as *aquatic*, *flying*, *amphibians*, and *terrestrial*, *etc*.
- All the camouflaged images are *hierarchically annotated* with category, bounding-box, object-level, and instance-level labels, facilitating many vision tasks, such as localization, object proposal, semantic edge detection [42], task transfer learning [69], *etc*.
- Each camouflaged image is assigned with *challenging attributes* found in the real-world and *matting-level* [73] labeling (requiring ∼60 minutes per image). These high-quality annotations could help with providing deeper insight into the performance of algorithms.

Second, using the collected *COD10K* and three existing datasets [56, 80, 80] we offer a rigorous evaluation of 13 state-of-the-art (SOTA) baselines [3, 23, 27, 35, 38, 40, 51, 68, 75, 77, 78, 83], making ours the largest COD study. Moreover, we propose a simple but efficient framework, named ***SINet*** (**S**earch and **I**dentification **Net**). Remarkably, the overall training time of *SINet* is only ∼1 hour and it achieves SOTA performance on all existing COD datasets, suggesting that it could be a potential solution to COD. Our work forms the first complete benchmark for the COD task in the deep learning era, bringing a novel view to object detection from a camouflage perspective.

2. Related Work

As suggested in [79], objects can be roughly divided into three categories: *generic objects*, *salient objects*, and *camouflaged objects*. We describe detection strategies for each type as follows.

2.1. Generic and Salient Object Detection

Generic Object Detection (GOD). One of the most popular directions in computer vision is generic object detection [11, 30, 37, 55]. Note that generic objects can be either

Dataset	Year	#Img.	#Cls.	Att.	BBox.	Ml.	Ins.	Cate.	Spi.	Obj.
CHAMELEON [56]	2018	76	-							✓
CPD1K [80]	2018	10,00	2							✓
CAMO [32]	2019	2,500	8	✓					✓	✓
***COD10K* (Ours)**	2020	**10,000**	**78**	✓	✓	✓	✓	✓	✓	✓

Table 1: Summary of COD datasets, showing *COD10K* offers much richer labels. Img.: Image. Cls.: Class. Att.: Attributes. BBox.: Bounding box. Ml.: Alpha-matting [73] level annotation (Fig. 7). Ins.: Instance. Cate.: Category. Obj.: Object. Spi.: Explicitly split the Training and Testing Set.

salient or camouflaged; camouflaged objects can be seen as difficult cases (the 2^{nd} and 3^{rd} row in Fig. 9) of generic objects. Typical GOD tasks include semantic segmentation and panoptic segmentation (see Fig. 2 b).

Salient Object Detection (SOD). This task aims to identify the most attention-grabbing object(s) in an image and then segment their pixel-level silhouettes [28, 72, 77]. Although the term "salient" is essentially the opposite of "camouflaged" (*standout vs. immersion*), salient objects can nevertheless provide important information for camouflaged object detection, *e.g.* by using images containing salient objects as the negative samples.

2.2. Camouflaged Object Detection

Research into camouflaged objects detection, which has had a tremendous impact on advancing our knowledge of visual perception, has a long and rich history in biology and art. Two remarkable studies on camouflaged animals from Abbott Thayer [58] and Hugh Cott [8] are still hugely influential. The reader can refer to Stevens *et al.*'s survey [57] for more details about this history.

Datasets. CHAMELEON [56] is an unpublished dataset that has only 76 images with manually annotated object-level ground-truths (GTs). The images were collected from the Internet via the Google search engine using "camouflaged animal" as a keyword. CPD1K [80] is the earliest dataset for camouflaged people detection, with 1K images covering two scene types, namely woodland and snowfield. Another contemporary dataset is CAMO [32], which has 2.5K images (2K for training, 0.5K for testing) covering eight categories. It has two sub-dataset, CAMO and MS-COCO, each of which contains 1.25K images.

Unlike existing datasets, the goal of our *COD10K* is to

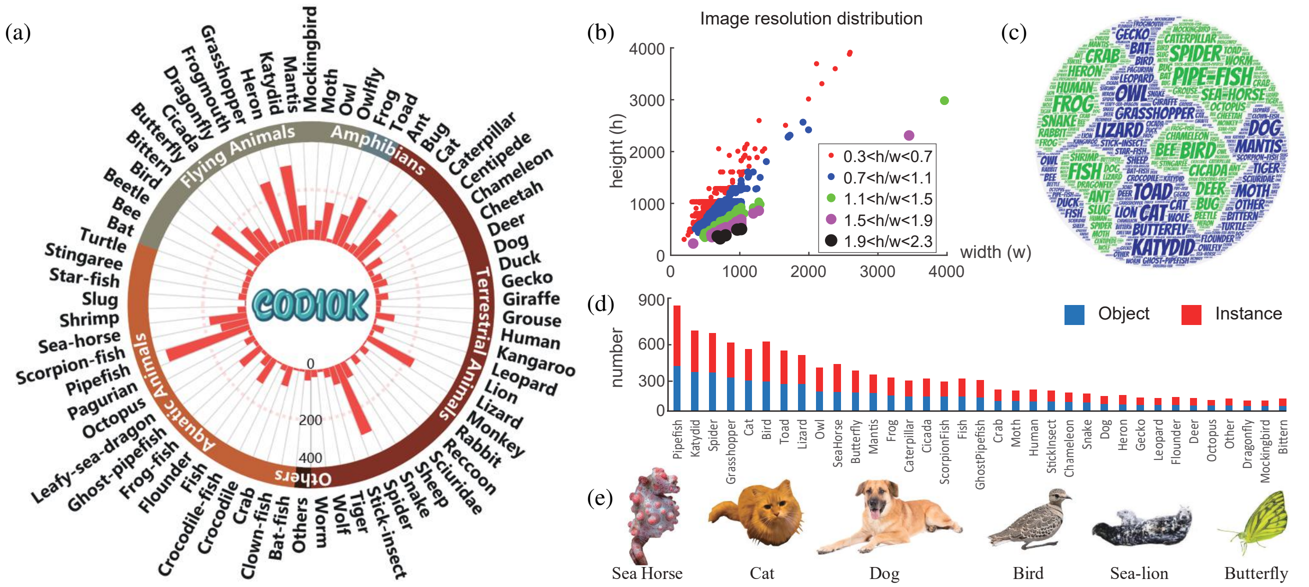


Figure 4: Statistics and camouflaged category examples from *COD10K* dataset. (a) Taxonomic system and its histogram distribution. (b) Image resolution distribution. (c) Word cloud distribution. (d) Object/Instance number of several categories. (e) Examples of sub-classes.

provide a more challenging, higher quality, and densely annotated dataset. *COD10K* is the largest camouflaged object detection dataset so far, containing 10K images (6K for training, 4K for testing). See Tab. 1 for details.

Types of Camouflage. Camouflaged images can be roughly split into two types: those containing natural camouflage and those with artificial camouflage. Natural camouflage is used by animals (*e.g.*, insects, cephalopods) as a survival skill to avoid recognition by a predator. In contrast, artificial camouflage is usually occurs in products (so-called defects) during the manufacturing process, or is used in gaming/art to hide information.

COD Formulation. Unlike class-dependent tasks such as semantic segmentation, COD is a class-independent task. Thus, the formulation of COD is simple and easy to define. Given an image, the task requires a *camouflaged object detection approach* to assign each pixel i a confidence $p_i \in$ [0,1], where p_i denotes the probability score of pixel i. A score of 0 is given to pixels that don't belong to the camouflaged objects, while a score of 1 indicates that a pixel is fully assigned to the camouflaged objects. This paper focuses on the object-level COD task, leaving instance-level COD to our future work.

Evaluation Metrics. Mean absolute error (MAE) is widely used in SOD tasks. Following Perazzi *et al*. [49], we also adopt the MAE (M) metric to assess the pixel-level accuracy between a predicted map C and ground-truth G. However, while useful for assessing the presence and amount of error, the MAE metric is not able to determine where the error occurs. Recently, Fan *et al*. proposed a human visual perception based E-measure (E_ϕ) [13], which simultaneously evaluates the pixel-level matching and image-level statistics. This metric is naturally suited for assessing the overall and localized accuracy of the camouflaged object detection results. Since camouflaged objects often contain complex shapes, COD also requires a metric that can judge structural similarity. We utilize the S-measure (S_α) [12] as our alternative metric. Recent studies [12, 13] have suggested that the weighted F-measure (F_β^w) [43] can provide more reliable evaluation results than the traditional F_β; thus, we also consider this metric in the COD field.

3. Proposed Dataset

The emergence of new tasks and datasets [7, 11, 36, 47, 82] has led to rapid progress in various areas of computer vision. For instance, ImageNet [52] revolutionized the use of deep models for visual recognition. With this in mind, our goals for studying and developing a dataset for COD are: (1) to provide a new challenging task, (2) to promote research in a new topic, and (3) to spark novel ideas. Exemplars of *COD10K* are shown in Fig. 1&3, and Fig. 4 (e). We will describe the details of *COD10K* in terms of three key aspects, as follows. The COD10K is available at here.

3.1. Image Collection

As suggested by [17, 50], the quality of annotation and size of a dataset are determining factors for its lifespan as a benchmark. To this end, *COD10K* contains 10,000 images (5,066 camouflaged, 3,000 background, 1,934 non-camouflaged), divided into 10 super-classes, and 78 sub-classes (69 camouflaged, nine non-camouflaged) which are collected from multiple photography websites.

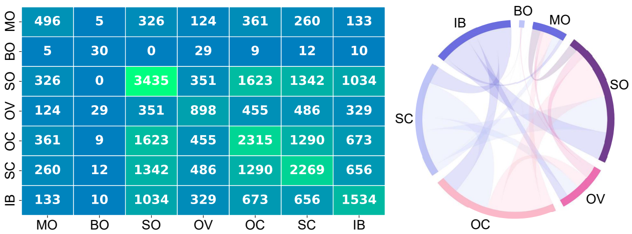


Figure 5: Left: Co-attributes distribution over *COD10K*. The number in each grid indicates the total number of images. Right: Multi-dependencies among these attributes. A larger arc length indicates a higher probability of one attribute correlating to another.

Attr	Description
MO	*Multiple Objects.* Image contains at least two objects.
BO	*Big Object.* Ratio (τ_{bo}) between object area and image area $\geq$0.5.
SO	*Small Object.* Ratio (τ_{so}) between object area and image area $\leq$0.1.
OV	*Out-of-View.* Object is clipped by image boundaries.
OC	*Occlusions.* Object is partially occluded.
SC	*Shape Complexity.* Object contains thin parts (*e.g.*, animal foot).
IB	*Indefinable Boundaries.* The foreground and background areas around the object have similar colors (χ^2 distance τ_{gc} between RGB histograms less than 0.9).

Table 2: Attribute descriptions (see examples in Fig. 3).

Most camouflaged images are from Flicker and have been applied for academic use with the following keywords: *camouflaged animal*, *unnoticeable animal*, *camouflaged fish*, *camouflaged butterfly*, *hidden wolf spider*, *walking stick*, *dead-leaf mantis*, *bird*, *sea horse*, *cat*, *pygmy seahorses*, *etc*. (see Fig. 4 e) The remaining camouflaged images (around 200 images) come from other websites, including Visual Hunt, Pixabay, Unsplash, Free-images, *etc*., which release public-domain stock photos, free from copyright and loyalties. To avoid selection bias [17], we also collected 3,000 salient images from Flickr. To further enrich the negative samples, 1,934 non-camouflaged images, including *forest*, *snow*, *grassland*, *sky*, *seawater* and other categories of background scenes, were selected from the Internet. For more details on the image selection scheme, we refer to Zhou *et al*. [81].

3.2. Professional Annotation

Recently released datasets [10, 15, 16] have shown that establishing a taxonomic system is crucial when creating a large-scale dataset. Motivated by [45], our annotations (obtained via crowdsourcing) are hierarchical (category $\rightarrowtail$ bounding box $\rightarrowtail$ attribute $\rightarrowtail$ object/instance).

• *Categories.* As illustrated in Fig. 4 (a), we first create five super-class categories. Then, we summarize the 69 most frequently appearing sub-class categories according to our collected data. Finally, we label the sub-class and super-class of each image. If the candidate image doesn't belong to any established category, we classify it as 'other'.

• *Bounding boxes.* To extend *COD10K* for the camouflaged object proposal task, we also carefully annotate the bounding boxes for each image.

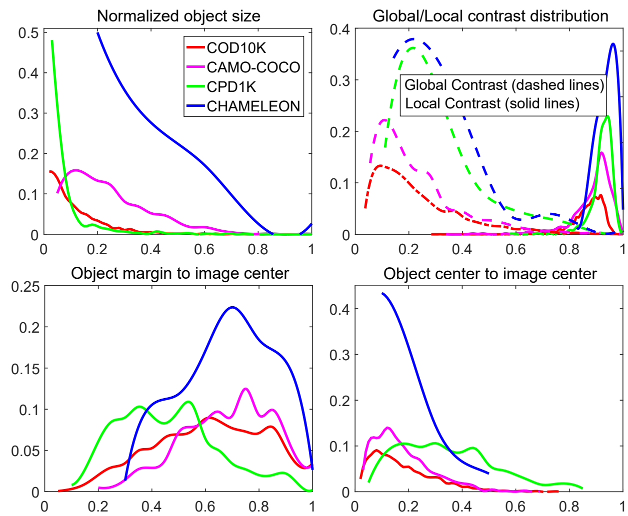


Figure 6: Comparison between the proposed *COD10K* and existing datasets. *COD10K* has smaller objects (top-left), contains more difficult camouflage (top-right), and suffers from less center bias (bottom-left/right).

• *Attributes.* In line with the literature [17, 50], we label each camouflaged image with highly challenging attributes faced in natural scenes, *e.g.*, *occlusions*, *indefinable boundaries*. Attribute descriptions are provided in Tab. 2, and the co-attribute distribution is shown in Fig. 5.

• *Objects/Instances.* We stress that existing COD datasets focus exclusively on object-level labels (Tab. 1). However, being able to parse an object into its instances is important for computer vision researchers to be able to edit and understand a scene. To this end, we further annotate objects at an instance-level, like COCO [36], resulting in 5,069 object-level masks and 5,930 instance-level GTs.

3.3. Dataset Features and Statistics

• *Object size.* Following [17], we plot the normalized object size in Fig. 6 (top-left), *i.e.*, the size distribution from 0.01%$\sim$ 80.74% (avg.: 8.94%), showing a broader range compared to CAMO-COCO, CPD1K, and CHAMELEON.

• *Global/Local contrast.* To evaluate whether an object is easy to detect, we describe it using the global/local contrast strategy [34]. Fig. 6 (top-right) shows that objects in *COD10K* are more challenging than those in other datasets.

• *Center bias.* This commonly occurs when taking a photo, as humans are naturally inclined to focus on the center of a scene. We adopt the strategy described in [17] to analyze this bias. Fig. 6 (bottom) shows that our dataset suffers from less center bias than others.

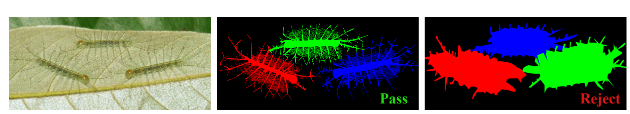


Figure 7: Alpha-matting [73] for high-quality annotation.

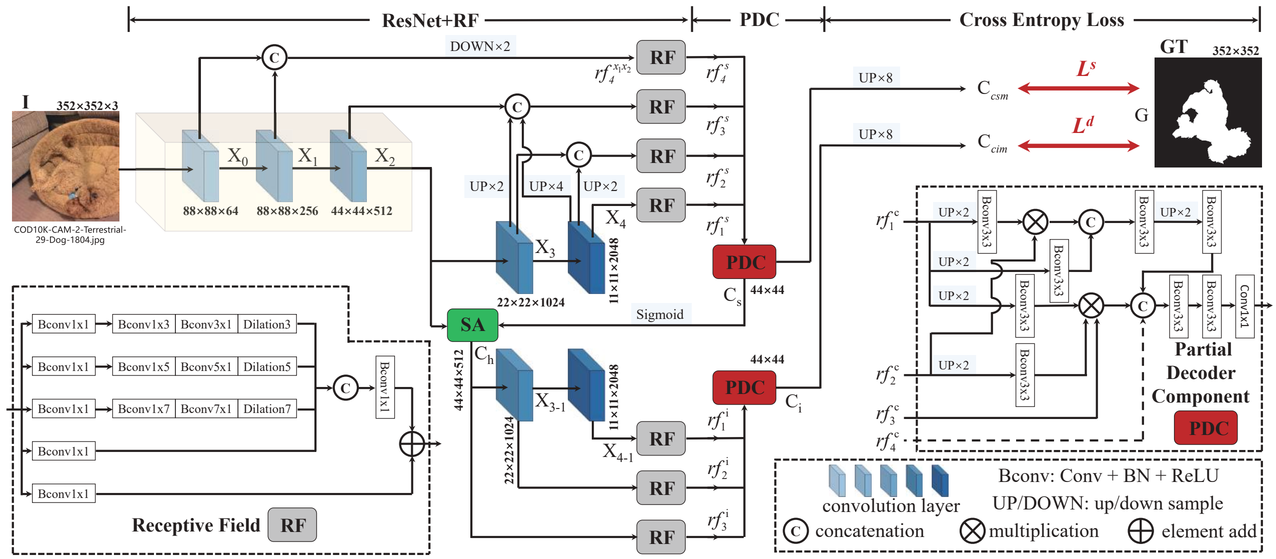


Figure 8: Overview of our *SINet* framework, which consists of two main components: the receptive field (RF) and partial decoder component (PDC). The RF is introduced to mimic the structure of RFs in the human visual system. The PDC reproduces the search and identification stages of animal predation. SA = search attention function described in [68]. See § 4 for details.

• *Quality control.* To ensure high-quality annotation, we invited three viewers to participate in the labeling process for 10-fold cross-validation. Fig. 7 shows examples that were passed/rejected. This instance-level annotation costs $\sim$ 60 minutes per image on average.

• *Super/Sub-class distribution.* *COD10K* includes five super-classes (*terrestrial*, *atmobios*, *aquatic*, *amphibian*, *other*) and 69 sub-classes (*e.g.*, *bat-fish*, *lion*, *bat*, *frog*, *etc*.). Examples of the wordcloud and object/instance number for various categories are shown in Fig. 4 c&d, respectively.

• *Resolution distribution.* As noted in [70], high-resolution data provides more object boundary details for model training and yields better performance when testing. Fig. 4 (b) presents the resolution distribution of *COD10K*, which includes a large number of Full HD 1080p images.

• *Dataset splits.* To provide a large amount of training data for deep learning models, *COD10K* is split into 6,000 images for training and 4,000 for testing, randomly selected from each sub-class.

4. Proposed Framework

Motivation. Biological studies [22] have shown that, when hunting, a predator will first judge whether a potential prey exists, *i.e.*, it will *search* for a prey; then, the target animal can be *identified*; and, finally, it can be *caught*.

Overview. The proposed *SINet* framework is inspired by the first two stages of hunting. It includes two main modules: the search module (SM) and the identification module (IM). The former (§ 4.1) is responsible for searching for a camouflaged object, while the latter (§ 4.2) is then used to precisely detect it.

4.1. Search Module (SM)

Neuroscience experiments have verified that, in the human visual system, a set of various sized population Receptive Fields (pRFs) helps to highlight the area close to the retinal fovea, which is sensitive to small spatial shifts [41]. This motivates us to use an RF [41,68] component to incorporate more discriminative feature representations during the searching stage (usually in a small/local space). Specifically, for an input image $\mathbf{I} \in \mathbb{R}^{W\times H\times 3}$, a set of features $\{\mathcal{X}_k\}_{k=0}^{4}$ is extracted from ResNet-50 [24]. To retain more information, we modify the parameter of stride = 1 to have the same resolution in the second layer. Thus, the resolution of each layer is $\{[\frac{H}{k}, \frac{W}{k}], k = 4, 4, 8, 16, 32\}$.

Recent evidence [78] has shown that low-level features in shallow layers preserve spatial details for constructing object boundaries, while high-level features in deep layers retain semantic information for locating objects. Due to this inherent property of neural networks, we divide the extracted features into low-level $\{\mathcal{X}_0, \mathcal{X}_1\}$, middle-level $\mathcal{X}_2$, high-level $\{\mathcal{X}_3, \mathcal{X}_4\}$ and combine them though concatenation, up-sampling, and down-sampling operations. Unlike [78], our *SINet* leverages a densely connected strategy [26] to preserve more information from different layers and then uses the modified RF [41] component to enlarge the receptive field. For example, we fuse the low-level features $\{\mathcal{X}_0, \mathcal{X}_1\}$ using a concatenation operation and then down-sample the resolution by half. This new feature rf_4^{x1x2} is then further fed into the RF component to generate the output feature rf_4^s. As shown in Fig. 8, after combining the three levels of features, we have a set of enhanced features $\{rf_k^s, k = 1, 2, 3, 4\}$ for learning robust cues.

Receptive Field (RF). The RF component includes five branches $\{b_k, k = 1, \ldots, 5\}$. In each branch, the first convolutional (Bconv) layer has dimensions 1×1 to reduce the channel size to 32. This is followed by two other layers: a $(2k-1) \times (2k-1)$ Bconv layer and a 3×3 Bconv layer with a specific dilation rate $(2k-1)$ when $k > 2$. The first four branches are concatenated and then their channel size is reduced to 32 with a 1×1 Bconv operation. Finally, the 5^{th} branch is added in and the whole module is fed to a ReLU function to obtain the feature rf_k.

4.2. Identification Module (IM)

After obtaining the candidate features from the previous search module, in the identification module, we need to precisely detect the camouflaged object. We extend the partial decoder component (PDC) [68] with a densely connected feature. More specifically, the PDC integrates four levels of features from SM. Thus, the coarse camouflage map C_s can be computed by

$$C_s = PD_s(rf_1^s, rf_2^s, rf_3^s, rf_4^s), \tag{1}$$

where $\{rf_k^s = rf_k, k = 1, 2, 3, 4\}$. Existing literature [40, 68] has shown that attention mechanisms can effectively eliminate interference from irrelevant features. We introduce a *search attention (SA)* module to enhance the middle-level features $\mathcal{X}_2$ and obtain the enhanced camouflage map C_h:

$$C_h = f_{max}(g(\mathcal{X}_2, \sigma, \lambda), C_s), \tag{2}$$

where $g(\cdot)$ is the SA function, which is actually a typical Gaussian filter with standard deviation $\sigma = 32$ and kernel size $\lambda = 4$, followed by a normalization operation. $f_{max}(\cdot)$ is a maximum function that highlights the initial camouflage regions of C_s.

To holistically obtain the high-level features, we further utilize PDC to aggregate another three layers of features, enhanced by the RF function, and obtain our final camouflage map C_i

$$C_i = PD_i(rf_1^i, rf_2^i, rf_3^i), \tag{3}$$

where $\{rf_k^i = rf_k, k = 1, 2, 3\}$. The difference between PD_s and PD_i is the number of input features.

Partial Decoder Component (PDC). Formally, given features $\{rf_k^c, k \in [m, \ldots, M], c \in [s, i]\}$ from the search and identification stages, we generate new features $\{rf_k^{c1}\}$ using the context module. Element-wise multiplication is adopted to decrease the gap between adjacent features. Specifically, for the shallowest feature, *e.g.*, rf_4^s, we set $rf_M^{c1} = rf_M^{c2}$ when $k = M$. For the deeper feature, *e.g.*, $rf_k^{c1}, k < M$, we update it as rf_k^{c2}:

$$rf_k^{c2} = rf_k^{c1} \otimes \Pi_{j=k+1}^{M} Bconv(UP(f_j^{c1})), \tag{4}$$

where $k \in [m, \ldots, M-1]$, $Bconv(\cdot)$ is a sequential operation that combines a 3×3 convolution followed by batch normalization, and a ReLU function. $UP(\cdot)$ is an up-sampling operation with a 2^{j-k} ratio. Finally, we combine these discriminative features via a concatenation operation. Our loss function for training *SINet* is the cross entropy [77] loss L_{CE}. The total loss function L is:

$$L = L_{CE}^{s}(C_{csm}, G) + L_{CE}^{i}(C_{cim}, G), \tag{5}$$

where C_{csm} and C_{cim} are the two camouflaged object maps obtained after C_s and C_i are up-sampled to a resolution of 352×352.

4.3. Implementation Details.

SINet is implemented in PyTorch and trained with the Adam optimizer [29]. During the training stage, the batch size is set to 36, and the learning rate starts at 1e-4. The whole training time is only about 70 minutes for 30 epochs (early-stop strategy). The running time is measured on the platform of Intel® i9-9820X CPU @3.30GHz $\times$ 20 and TITAN RTX. The inference time is 0.2s for a 352×352 image.

5. Benchmark Experiments

5.1. Experimental Settings

Training/Testing Details. To verify the generalizability of *SINet*, we provide four training settings, using the training sets (camouflaged images) from: **(i)** CPD1K [80], **(ii)** CAMO [32] **(iii)** COD10K, and **(iv)** CPD1K + CAMO + COD10K. For CPD1K, we follow [80] and randomly select 60% of images as the training set. For CAMO, we use the default training set. For COD10K, we use the default training camouflaged images. We evaluate our model on the whole CHAMELEON [56] dataset and the test sets of CPD1K, CAMO, and COD10K.

Baselines. To the best of our knowledge, there is no deep network based COD model that is publicly available. We therefore select 12 deep learning baselines [3, 23, 27, 35, 38, 40, 51, 68, 75, 77, 78, 83] according to the following criteria: (1) classical architectures, (2) recently published, (3) achieve SOTA performance in a specific field, *e.g.*, GOD or SOD. These baselines are trained with the recommended parameter settings, using the (iv) training setting.

5.2. Results and Data Analysis

Performance on CHAMELEON. From Tab. 3, compared with the 12 SOTA object detection baselines, our *SINet* achieves the best performances across all metrics. Note that our model does not apply any auxiliary edge/boundary features (*e.g.*, EGNet [77], PFANet [78], PoolNet [38]), preprocessing techniques [46], or post-processing strategies (*e.g.*, CRF [31], graph cut [2]).

Performance on CPD1K. To investigate the performance of the algorithm on a particular type of camouflage, we further adopt the CPD1K [80] dataset, which only contains

Baseline Models	CHAMELEON [56]				CPD1K-Test [80]				CAMO-Test [32]				COD10K-Test (Ours)			
	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$
2017 FPN [35]	0.794	0.783	0.590	0.075	0.786	0.767	0.491	0.013	0.684	0.677	0.483	0.131	0.697	0.691	0.411	0.075
2017 MaskRCNN [23]	0.643	0.778	0.518	0.099	0.614	0.671	0.299	0.037	0.574	0.715	0.430	0.151	0.613	0.748	0.402	0.080
2017 PSPNet [75]	0.773	0.758	0.555	0.085	0.765	0.776	0.430	0.017	0.663	0.659	0.455	0.139	0.678	0.680	0.377	0.080
2018 UNet++ [83]	0.695	0.762	0.501	0.094	0.714	0.704	0.410	0.017	0.599	0.653	0.392	0.149	0.623	0.672	0.350	0.086
2018 PiCANet [40]	0.769	0.749	0.536	0.085	0.754	0.751	0.400	0.024	0.609	0.584	0.356	0.156	0.649	0.643	0.322	0.090
2019 MSRCNN [27]	0.637	0.686	0.443	0.091	0.743	0.789	0.545	0.010	0.617	0.669	0.454	0.133	0.641	0.706	0.419	0.073
2019 PoolNet [38]	0.776	0.779	0.555	0.081	0.632	0.712	0.245	0.026	0.702	0.698	0.494	0.129	0.705	0.713	0.416	0.074
2019 BASNet [51]	0.687	0.721	0.474	0.118	0.756	0.817	0.512	0.018	0.618	0.661	0.413	0.159	0.634	0.678	0.365	0.105
2019 PFANet [78]	0.679	0.648	0.378	0.144	0.658	0.645	0.224	0.042	0.659	0.622	0.391	0.172	0.636	0.618	0.286	0.128
2019 CPD [68]	0.853	0.866	0.706	0.052	0.828	0.829	0.579	0.010	0.726	0.729	0.550	0.115	0.747	0.770	0.508	0.059
2019 HTC [3]	0.517	0.489	0.204	0.129	0.703	0.767	0.452	0.019	0.476	0.442	0.174	0.172	0.548	0.520	0.221	0.088
2019 EGNet [77]	0.848	0.870	0.702	0.050	0.588	0.503	0.249	0.019	0.732	0.768	0.583	0.104	0.737	0.779	0.509	0.056
2019 ANet-SRM [32]	‡	‡	‡	‡	‡	‡	‡	‡	0.682	0.685	0.484	0.126	‡	‡	‡	‡
SINet'20 Training setting (i)	0.446	0.431	0.115	0.269	0.523	0.652	0.076	0.201	0.483	0.477	0.189	0.285	0.448	0.476	0.087	0.258
SINet'20 Training setting (ii)	0.737	0.737	0.478	0.103	0.633	0.693	0.190	0.068	0.708	0.706	0.476	0.131	0.685	0.718	0.352	0.092
SINet'20 Training setting (iii)	0.846	0.871	0.691	0.050	0.675	0.721	0.267	0.026	0.665	0.662	0.470	0.128	0.758	0.796	0.517	0.054
SINet'20 Training setting (iv)	**0.869**	**0.891**	**0.740**	**0.044**	**0.849**	**0.869**	**0.587**	**0.010**	**0.751**	**0.771**	**0.606**	**0.100**	**0.771**	**0.806**	**0.551**	**0.051**

Table 3: Quantitative results on different datasets. The best scores are highlighted in **bold**. See § 5.1 for training details: (i) CPD1K, (ii) CAMO, (iii) COD10K, (iv) CPD1K + CAMO + COD10K. Note that the ANet-SRM model (only trained on CAMO) does not have a publicly available code, thus other results are not available ('‡'). ↑ indicates the higher the score the better. E_ϕ denotes mean E-measure [13]. Baseline models are trained using the training setting (iv). Evaluation code: https://github.com/DengPingFan/CODToolbox

camouflaged person. Interestingly, we find that, while the strong EGNet baseline achieves good performance ($S_\alpha > 0.7$) on other datasets, its score drops dramatically ($S_\alpha = 0.588$) on CPD1K due to the large number of small objects. Specifically, our model outperforms the best CPD [68] baseline by ∼2% (S_α score).

Performance on CAMO. We also test our model on the recently proposed CAMO [32] dataset, which includes various camouflaged objects. Based on the overall performances reported in Tab. 3, we find that the CAMO dataset is more challenging than the previous two datasets (CHAMELEON, CPD1K). Again, *SINet* obtains the best performance, further demonstrating its robustness.

Performance on COD10K. With the test set (2,026 images) of our COD10K dataset, we again observe that the proposed *SINet* is consistently better than other competitors. This is because its specially designed search and identification modules can automatically learn rich high-/middle-/low-level features, which are crucial for overcoming challenging ambiguities in object boundaries (see Fig. 9).

GOD vs. SOD Baselines. One noteworthy finding is that, among the top-3 models, the GOD model (*i.e.*, FPN [35]) performs worse than the SOD competitors, CPD [68], EGNet [77], suggesting that the SOD framework may be better suited for extension to COD tasks. Compared with either the GOD [3, 23, 27, 35, 75, 83] or the SOD [38, 40, 51, 68, 77, 78] models, *SINet* significantly decreases the training time (*e.g.*, *SINet*: 1 hour vs. EGNet: 48 hours) and achieves the SOTA performance on all datasets, showing that it is a promising solution for the COD problem.

Cross-dataset Generalization. The generalizability and difficulty of datasets play a crucial role in both training and assessing different algorithms [61]. Hence, we study these

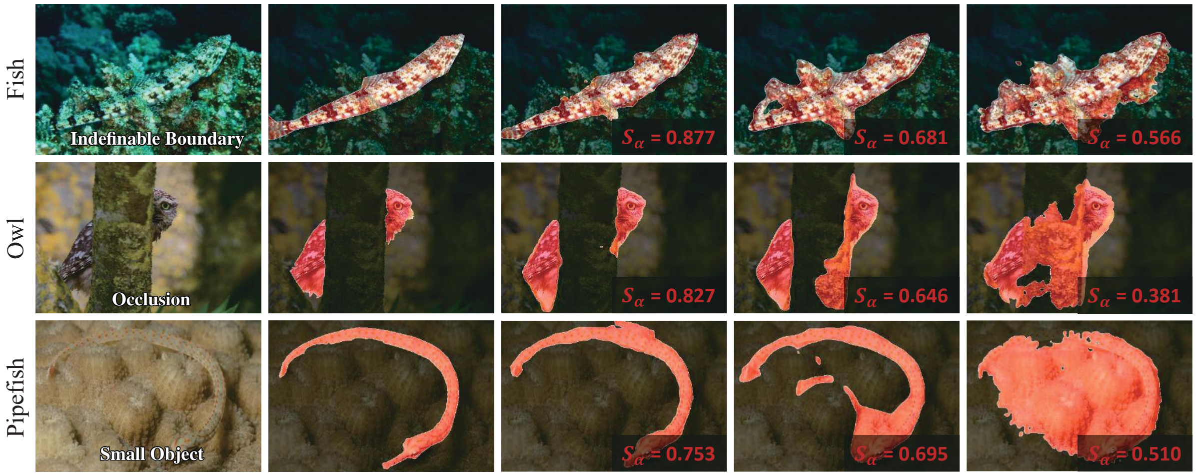
Trained on: \ Tested on:	CPD1K [80]	CAMO [32]	COD10K (Ours)	Self	Mean others	Drop↓
CPD1K [80]	**0.626**	0.540	0.475	0.626	0.508	18.8%
CAMO [32]	0.654	**0.803**	0.702	0.803	0.678	15.6%
COD10K (Ours)	0.624	0.742	**0.700**	0.700	0.683	2.40%
Mean others	0.639	0.641	0.589			

Table 4: S-measure↑ [12] results for cross-dataset generalization. *SINet* is trained on one (rows) dataset and tested on all datasets (columns). "Self": training and testing on the same (diagonal) dataset. "Mean others": average score on all except self.

aspects for existing COD datasets, using the cross-dataset analysis method [59], *i.e.*, training a model on one dataset, and testing it on others. We select three datasets, including CDP1K [80], CAMO [32], and our *COD10K*. Following [61], for each dataset, we randomly select 800 images as the training set and 200 images as the testing set, since CPD1K only contains 1,000 images. For fair comparison, we train *SINet* on each dataset until the loss is stable.

Tab. 4 provides the S-measure results for the cross-dataset generalization. Each row lists a model that is trained on one dataset and tested on all others, indicating the generalizability of the dataset used for training. Each column shows the performance of one model tested on a specific dataset and trained on all others, indicating the difficulty of the testing dataset. Please note that the training/testing settings are different from those used in Tab. 3, and thus the performances are not comparable. As expected, we find that our *COD10K* is the most difficult (*e.g.*, the last row *Mean others:* 0.589). This is because our dataset contains a variety of challenging camouflaged objects (see § 3). CPD1K [80] has the worst generalizability (last column *Drop* 18.8%), which would make an algorithm ranking 1^{st} on CPD1K drop to one of the worst on other datasets.

Qualitative Analysis. Fig. 9 presents qualitative compar-



(a) Image (b) GT (c) SINet (Ours) (d) EGNet [77] (e) PFANet [78]

Figure 9: Qualitative results of our *SINet* and two top-performing baselines on *COD10K*. Refer to the supplementary material for details.

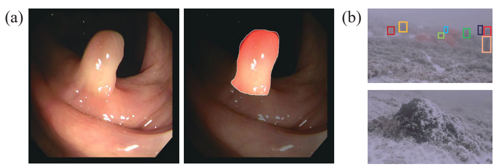


Figure 10: More applications. (a) Polyp detection/segmentation results. (b) Search and rescue system working in a disaster area.

isons between our *SINet* and two baselines. As can be seen, PFANet [78] is able to locate the camouflaged objects, but the outputs are always inaccurate. By further using edge features, EGNet [77] achieves a relatively more accurate location than PFANet. Nevertheless, it still misses the fine details of objects, especially for the *fish* in the 1^{st} row. For all these challenging cases (*e.g.*, *indefinable boundaries*, *occlusions*, and *small objects*), *SINet* is able to infer the real camouflaged object with fine details, demonstrating the robustness of our framework.

6. Potential Applications

Camouflage detection systems (CDS) have various possible applications. Here, we envision two potential uses.

Medical Image Segmentation. If a medical image segmentation method was equipped with a CDS trained for specific objects, such as polyp, it could be used to automatically segment polyps (Fig. 10 a), in nature to find & protect rare species, or even in disaster areas for search and rescue.

Search Engines. Fig. 11 shows an example of search results from Google. From the results (Fig. 11 a), we notice that the search engine cannot detect the concealed butterfly, and thus only provides images with similar backgrounds. Interestingly, when the search engine is equipped with a CDS (here, we just simply change the keyword), the engine can identify the camouflaged object and then feedback several butterfly images (Fig. 11 b).

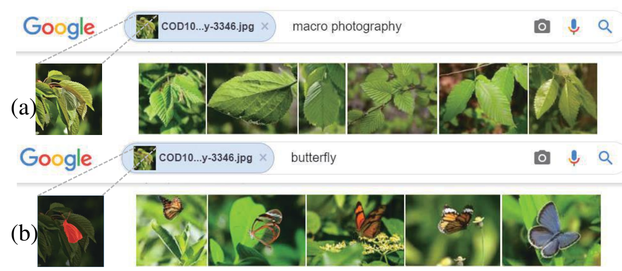


Figure 11: Search engine equipped without (a)/with (b) a CDS.

7. Conclusion

We have presented the first complete benchmark on object detection from a camouflage perspective. Specifically, we have provided a new challenging and densely annotated *COD10K* dataset, conducted a large-scale evaluation, developed a simple but efficient end-to-end *SINet* framework, and provided several potential applications. Compared with existing cutting-edge baselines, *SINet* is competitive and generates more visually favorable results. The above contributions offer the community an opportunity to design new models for the COD task. In future work, we plan to extend *COD10K* dataset to provide input of various forms, for example, RGB-D camouflage object detection (similar to RGB-D salient object detection [19, 71, 74]), among others. New techniques such as weakly supervised learning [53, 54], zero-shot learning [84], VAE [85], and multiscale backbone [20] could also be explored.

Acknowledgments. We thank Geng Chen and Hongsong Wang for insightful feedback. This research was supported by Major Project for New Generation of AI under Grant No. 2018AAA0100400, NSFC (61620106008), the Beijing Natural Science Foundation under Grant 4182056, the national youth talent support program, Zhejiang Lab's Open Fund under grant No. 2019KD0AB04, and Tianjin Natural Science Foundation (18ZXZNGX00110).

References

[1] Ali Borji, Ming-Ming Cheng, Qibin Hou, Huaizu Jiang, and Jia Li. Salient object detection: A survey. *Computational Visual Media*, 5(2):117–150, 2019.

[2] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. In *IEEE CVPR*, pages 377–384, 1999.

[3] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *IEEE CVPR*, pages 4974–4983, 2019.

[4] Ming-Ming Cheng, Yun Liu, Wen-Yan Lin, Ziming Zhang, Paul L Rosin, and Philip HS Torr. Bing: Binarized normed gradients for objectness estimation at 300fps. *Computational Visual Media*, 5(1):3–20, 2019.

[5] Ming-Ming Cheng, Niloy J. Mitra, Xiaolei Huang, Philip H. S. Torr, and Shi-Min Hu. Global contrast based salient region detection. *IEEE TPAMI*, 37(3):569–582, 2015.

[6] Hung-Kuo Chu, Wei-Hsin Hsu, Niloy J Mitra, Daniel Cohen-Or, Tien-Tsin Wong, and Tong-Yee Lee. Camouflage images. *ACM Trans. Graph.*, 29(4):51–1, 2010.

[7] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE CVPR*, pages 3213–3223, 2016.

[8] Hugh Bamford Cott. *Adaptive coloratcottion in animals*. Methuen & Co., Ltd., 1940.

[9] Innes C Cuthill, Martin Stevens, Jenna Sheppard, Tracey Maddocks, C Alejandro Párraga, and Tom S Troscianko. Disruptive coloration and background pattern matching. *Nature*, 434(7029):72, 2005.

[10] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The epic-kitchens dataset. In *ECCV*, pages 720–736, 2018.

[11] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes challenge: A retrospective. *IJCV*, 111(1):98–136, 2015.

[12] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A New Way to Evaluate Foreground Maps. In *IEEE ICCV*, pages 4548–4557, 2017.

[13] Deng-Ping Fan, Cheng Gong, Yang Cao, Bo Ren, Ming-Ming Cheng, and Ali Borji. Enhanced-alignment Measure for Binary Foreground Map Evaluation. In *IJCAI*, pages 698–704, 2018.

[14] Deng-Ping Fan, Ge-Peng Ji, Tao Zhou, Geng Chen, Huazhu Fu, Jianbing Shen, and Ling Shao. PraNet: Parallel Reverse Attention Network for Polyp Segmentation. *arXiv*, 2020.

[15] Deng-Ping Fan, Zheng Lin, Ge-Peng Ji, Dingwen Zhang, Huazhu Fu, and Ming-Ming Cheng. Taking a deeper look at the co-salient object detection. In *IEEE CVPR*, 2020.

[16] Deng-Ping Fan, Zheng Lin, Zhao Zhang, Menglong Zhu, and Ming-Ming Cheng. Rethinking RGB-D Salient Object Detection:Models, Datasets, and Large-Scale Benchmarks. *IEEE TNNLS*, 2020.

[17] Deng-Ping Fan, Jiang-Jiang Liu, Shang-Hua Gao, Qibin Hou, Ali Borji, and Ming-Ming Cheng. Salient objects in clutter: Bringing salient object detection to the foreground. In *ECCV*, pages 1597–1604. Springer, 2018.

[18] Deng-Ping Fan, Tao Zhou, Ge-Peng Ji, Yi Zhou, Geng Chen, Huazhu Fu, Jianbing Shen, and Ling Shao. Inf-Net: Automatic COVID-19 Lung Infection Segmentation from CT Scans. *arXiv preprint arXiv:2004.14133*, 2020.

[19] Keren Fu, Deng-Ping Fan, Ge-Peng Ji, and Qijun Zhao. JL-DCF: Joint Learning and Densely-Cooperative Fusion Framework for RGB-D Salient Object Detection. In *IEEE CVPR*, 2020.

[20] Shanghua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip HS Torr. Res2net: A new multi-scale backbone architecture. *IEEE TPMAI*, 2020.

[21] Shiming Ge, Xin Jin, Qiting Ye, Zhao Luo, and Qiang Li. Image editing by object-aware optimal boundary searching and mixed-domain composition. *CVM*, 4(1):71–82, 2018.

[22] Joanna R Hall, Innes C Cuthill, Roland Baddeley, Adam J Shohet, and Nicholas E Scott-Samuel. Camouflage, detection and identification of moving targets. *Proc. R. Soc. B: Biological Sciences*, 280(1758):20130064, 2013.

[23] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *IEEE ICCV*, pages 2961–2969, 2017.

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE CVPR*, pages 770–778, 2016.

[25] Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip Torr. Deeply supervised salient object detection with short connections. *IEEE TPAMI*, 41(4):815–828, 2019.

[26] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *IEEE CVPR*, pages 4700–4708, 2017.

[27] Zhaojin Huang, Lichao Huang, Yongchao Gong, Chang Huang, and Xinggang Wang. Mask scoring r-cnn. In *IEEE CVPR*, pages 6409–6418, 2019.

[28] Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE TPAMI*, 20(11):1254–1259, 1998.

[29] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[30] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *IEEE CVPR*, pages 9404–9413, 2019.

[31] Philipp Krahenbuhl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NIPS*, pages 109–117, 2011.

[32] Trung-Nghia Le, Tam V Nguyen, Zhongliang Nie, Minh-Triet Tran, and Akihiro Sugimoto. Anabranch network for camouflaged object segmentation. *CVIU*, 184:45–56, 2019.

[33] Guanbin Li, Yuan Xie, Liang Lin, and Yizhou Yu. Instance-level salient object segmentation. In *IEEE CVPR*, pages 247–256, 2017.

[34] Yin Li, Xiaodi Hou, Christof Koch, James M Rehg, and Alan L Yuille. The secrets of salient object segmentation. In *IEEE CVPR*, pages 280–287, 2014.

[35] Tsungyi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE CVPR*, pages 936–944, 2017.

[36] 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.

[37] Ce Liu, Jenny Yuen, and Antonio Torralba. Sift flow: Dense correspondence across scenes and its applications. *IEEE TPAMI*, 33(5):978–994, 2010.

[38] Jiang-Jiang Liu, Qibin Hou, Ming-Ming Cheng, Jiashi Feng, and Jianmin Jiang. A simple pooling-based design for real-time salient object detection. *IEEE CVPR*, 2019.

[39] Li Liu, Wanli Ouyang, Xiaogang Wang, Paul Fieguth, Jie Chen, Xinwang Liu, and Matti Pietikäinen. Deep learning for generic object detection: A survey. *IJCV*, 2019.

[40] Nian Liu, Junwei Han, and Ming-Hsuan Yang. Picanet: Learning pixel-wise contextual attention for saliency detection. In *IEEE CVPR*, pages 3089–3098, 2018.

[41] Songtao Liu, Di Huang, et al. Receptive field block net for accurate and fast object detection. In *ECCV*, pages 385–400, 2018.

[42] Yun Liu, Ming-Ming Cheng, Deng-Ping Fan, Le Zhang, JiaWang Bian, and Dacheng Tao. Semantic edge detection with diverse deep supervision. *arXiv preprint arXiv:1804.02864*, 2018.

[43] Ran Margolin, Lihi Zelnik-Manor, and Ayellet Tal. How to evaluate foreground maps? In *IEEE CVPR*, pages 248–255, 2014.

[44] Gerard Medioni. Generic object recognition by inference of 3-d volumetric. *Object Categorization: Computer and Human Vision Perspectives*, 87, 2009.

[45] Kaichun Mo, Shilin Zhu, Angel X Chang, Li Yi, Subarna Tripathi, Leonidas J Guibas, and Hao Su. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *IEEE CVPR*, pages 909–918, 2019.

[46] Greg Mori. Guiding model search using segmentation. In *IEEE ICCV*, pages 1417–1423, 2005.

[47] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *IEEE CVPR*, pages 4990–4999, 2017.

[48] Andrew Owens, Connelly Barnes, Alex Flint, Hanumant Singh, and William Freeman. Camouflaging an object from many viewpoints. In *IEEE CVPR*, pages 2782–2789, 2014.

[49] Federico Perazzi, Philipp Krähenbühl, Yael Pritch, and Alexander Hornung. Saliency filters: Contrast based filtering for salient region detection. In *IEEE CVPR*, pages 733–740, 2012.

[50] 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 *IEEE CVPR*, pages 724–732, 2016.

[51] Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jagersand. Basnet: Boundary-aware salient object detection. In *IEEE CVPR*, pages 7479–7489, 2019.

[52] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, and et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015.

[53] Yunhang Shen, Rongrong Ji, Yan Wang, Yongjian Wu, and Liujuan Cao. Cyclic guidance for weakly supervised joint detection and segmentation. In *IEEE CVPR*, pages 697–707, 2019.

[54] Yunhan Shen, Rongrong Ji, Shengchuan Zhang, Wangmeng Zuo, and Yan Wang. Generative adversarial learning towards fast weakly supervised detection. In *IEEE CVPR*, pages 5764–5773, 2018.

[55] Jamie Shotton, John Winn, Carsten Rother, and Antonio Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *ECCV*, pages 1–15. Springer, 2006.

[56] P Skurowski, H Abdulameer, J Błaszczyk, T Depta, A Kornacki, and P Kozieł. Animal camouflage analysis: Chameleon database. Unpublished Manuscript, 2018.

[57] Martin Stevens and Sami Merilaita. Animal camouflage: current issues and new perspectives. *Phil. Trans. R. Soc. B: Biological Sciences*, 364(1516):423–427, 2008.

[58] Gerald Handerson Thayer and Abbott Handerson Thayer. *Concealing-coloration in the Animal Kingdom: An Exposition of the Laws of Disguise Through Color and Pattern: Being a Summary of Abbott H. Thayer's Discoveries*. Macmillan Company, 1909.

[59] Antonio Torralba, Alexei A Efros, et al. Unbiased look at dataset bias. In *IEEE CVPR*, pages 1521–1528, 2011.

[60] Tom Troscianko, Christopher P Benton, P George Lovell, David J Tolhurst, and Zygmunt Pizlo. Camouflage and visual perception. *Phil. Trans. R. Soc. B: Biological Sciences*, 364(1516):449–461, 2008.

[61] Wenguan Wang, Qiuxia Lai, Huazhu Fu, Jianbing Shen, and Haibin Ling. Salient object detection in the deep learning era: An in-depth survey. *arXiv preprint arXiv:1904.09146*, 2019.

[62] Wenguan Wang and Jianbing Shen. Deep visual attention prediction. *IEEE TIP*, 27(5):2368–2378, 2017.

[63] Wenguan Wang, Jianbing Shen, Ming-Ming Cheng, and Ling Shao. An iterative and cooperative top-down and bottom-up inference network for salient object detection. In *IEEE CVPR*, pages 5968–5977, 2019.

[64] Wenguan Wang, Jianbing Shen, Xingping Dong, and Ali Borji. Salient object detection driven by fixation prediction. In *IEEE CVPR*, pages 1711–1720, 2018.

[65] Wenguan Wang, Jianbing Shen, Ling Shao, and Fatih Porikli. Correspondence driven saliency transfer. *IEEE TIP*, 25(11):5025–5034, 2016.

[66] Wenguan Wang, Shuyang Zhao, Jianbing Shen, Steven CH Hoi, and Ali Borji. Salient object detection with pyramid attention and salient edges. In *IEEE CVPR*, pages 1448–1457, 2019.

[67] Yu-Huan Wu, Shang-Hua Gao, Jie Mei, Jun Xu, Deng-Ping Fan, Chao-Wei Zhao, and Ming-Ming Cheng. JCS: An Explainable COVID-19 Diagnosis System by Joint Classification and Segmentation. *arXiv preprint arXiv:2004.07054*, 2020.

[68] Zhe Wu, Li Su, and Qingming Huang. Cascaded partial decoder for fast and accurate salient object detection. In *IEEE CVPR*, pages 3907–3916, 2019.

[69] Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *IEEE CVPR*, pages 3712–3722, 2018.

[70] Yi Zeng, Pingping Zhang, Jianming Zhang, Zhe Lin, and Huchuan Lu. Towards high-resolution salient object detection. In *IEEE ICCV*, 2019.

[71] Jing Zhang, Deng-Ping Fan, Yuchao Dai, Saeed Anwar, Fatemeh Sadat Saleh, Tong Zhang, and Nick Barnes. UC-Net: Uncertainty Inspired RGB-D Saliency Detection via Conditional Variational Autoencoders. In *IEEE CVPR*, 2020.

[72] Pingping Zhang, Dong Wang, Huchuan Lu, Hongyu Wang, and Xiang Ruan. Amulet: Aggregating multi-level convolutional features for salient object detection. In *IEEE CVPR*, pages 202–211, 2017.

[73] Yunke Zhang, Lixue Gong, Lubin Fan, Peiran Ren, Qixing Huang, Hujun Bao, and Weiwei Xu. A late fusion cnn for digital matting. In *IEEE CVPR*, pages 7469–7478, 2019.

[74] Zhao Zhang, Zheng Lin, Jun Xu, Wenda Jin, Shao-Ping Lu, and Deng-Ping Fan. Bilateral attention network for rgb-d salient object detection. *arXiv preprint arXiv:2004.14582*, 2020.

[75] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *IEEE CVPR*, pages 6230–6239, 2017.

[76] Jia-Xing Zhao, Yang Cao, Deng-Ping Fan, Ming-Ming Cheng, Xuan-Yi Li, and Le Zhang. Contrast prior and fluid pyramid integration for RGBD salient object detection. In *IEEE CVPR*, pages 3927–3936, 2019.

[77] Jia-Xing Zhao, Jiang-Jiang Liu, Deng-Ping Fan, Yang Cao, Jufeng Yang, and Ming-Ming Cheng. Egnet:edge guidance network for salient object detection. In *IEEE ICCV*, 2019.

[78] Ting Zhao and Xiangqian Wu. Pyramid feature attention network for saliency detection. In *IEEE CVPR*, pages 3085–3094, 2019.

[79] Zhong-Qiu Zhao, Peng Zheng, Shou-tao Xu, and Xindong Wu. Object detection with deep learning: A review. *IEEE TNNLS*, 30(11):3212–3232, 2019.

[80] Yunfei Zheng, Xiongwei Zhang, Feng Wang, Tieyong Cao, Meng Sun, and Xiaobing Wang. Detection of people with camouflage pattern via dense deconvolution network. *IEEE SPL*, 26(1):29–33, 2018.

[81] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE TPAMI*, 40(6):1452–1464, 2017.

[82] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *IEEE CVPR*, pages 633–641, 2017.

[83] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In *DLMIA*, pages 3–11, 2018.

[84] Yizhe Zhu, Mohamed Elhoseiny, Bingchen Liu, Xi Peng, and Ahmed Elgammal. A generative adversarial approach for zero-shot learning from noisy texts. In *CVPR*, pages 1004–1013, 2018.

[85] Yizhe Zhu, Martin Renqiang Min, Asim Kadav, and Hans Peter Graf. S3VAE: Self-Supervised Sequential VAE for Representation Disentanglement and Data Generation. In *CVPR*, 2020.