Computer Vision and Image Understanding 184 (2019) 45–56

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Anabranch network for camouflaged object segmentation

Trung-Nghia Le [a,1], Tam V. Nguyen [b,*], Zhongliang Nie [b], Minh-Triet Tran [c], Akihiro Sugimoto [d]

[a] *Department of Informatics, The Graduate University for Advanced Studies (SOKENDAI), Tokyo, Japan*
[b] *Department of Computer Science, University of Dayton, OH, United States*
[c] *University of Science, VNU-HCM, Ho Chi Minh, Vietnam*
[d] *National Institute of Informatics, Tokyo, Japan*

ARTICLE INFO

Communicated by N. Paragios

Keywords:
Camouflaged object segmentation
Anabranch network

ABSTRACT

Camouflaged objects attempt to conceal their texture into the background and discriminating them from the background is hard even for human beings. The main objective of this paper is to explore the camouflaged object segmentation problem, namely, segmenting the camouflaged object(s) for a given image. This problem has not been well studied in spite of a wide range of potential applications including the preservation of wild animals and the discovery of new species, surveillance systems, search-and-rescue missions in the event of natural disasters such as earthquakes, floods or hurricanes. This paper addresses a new challenging problem of camouflaged object segmentation. To address this problem, we provide a new image dataset of camouflaged objects for benchmarking purposes. In addition, we propose a general end-to-end network, called the Anabranch Network, that leverages both classification and segmentation tasks. Different from existing networks for segmentation, our proposed network possesses the second branch for classification to predict the probability of containing camouflaged object(s) in an image, which is then fused into the main branch for segmentation to boost up the segmentation accuracy. Extensive experiments conducted on the newly built dataset demonstrate the effectiveness of our network using various fully convolutional networks.

1. Introduction

Camouflage is an attempt to conceal the texture of a foreground object into the background (Singh et al., 2013). The term “*camouflage*” was first coined from nature where animals used to hide themselves from predators by changing their body pattern, texture, or color. Unlike salient objects, camouflaged objects are hard to detect in their nature even for human beings. Autonomously detecting/segmenting camouflaged objects is thus a challenging task where discriminative features do not play an important role anymore since we have to ignore objects that capture our attention. While detecting camouflaged objects is technically difficult on one hand, it is beneficial in various practical scenarios, on the other hand, which include surveillance systems, and search-and-rescue missions. For example, detecting and segmenting camouflaged objects in images definitely helps us search for camouflaged animals for the preservation of wild animals and the discovery of new species. Nevertheless, camouflaged object segmentation has not been well explored in the literature.

There exist two types of camouflaged objects in reality: naturally camouflaged objects such as animals or insects hiding themselves from their predators, and artificially camouflaged objects such as soldiers and weapons disguised by artificial texture patterns. These objects first evaluate their surrounding environment and then change to camouflaged textures via the colorization of cloths or cover. For either type of camouflaged objects, it is not obvious to identify them in images. Fig. 1 shows a few examples of camouflaged objects in real life. Based on this figure, we can easily see how challenging camouflaged object segmentation is.

Camouflaged objects have naturally evolved to exploit weaknesses in the visual system of their prey or predator, and thus understanding such a mechanism will provide some insights for segmenting camouflaged objects. Without any prior, we human beings easily miss detecting camouflaged objects; however, once we are informed a camouflaged object exists in an image, we can carefully scan the entire image to detect it. This also comes up in nature: once the predator has an awareness of camouflaged animals in the scene, then it makes efforts to localize them for hunting. Therefore, utilizing the awareness as a prior can be an essential cue for camouflaged object segmentation. Then, the challenge is what awareness should be incorporated in what way. We consider that classification scheme can be used as the awareness of existing of camouflaged objects in an image in order to combine with detection/segmentation scheme.

[*] Corresponding author.
E-mail addresses: ltnghia@its.iis.u-tokyo.ac.jp (T.-N. Le), tamnguyen@udayton.edu (T.V. Nguyen).
[1] Trung-Nghia Le contributed to this work while he interned at University of Dayton. He is currently with the Institute of Industrial Science, the University of Tokyo, Japan.

https://doi.org/10.1016/j.cviu.2019.04.006
Received 21 July 2018; Received in revised form 29 January 2019; Accepted 17 April 2019
Available online 2 May 2019

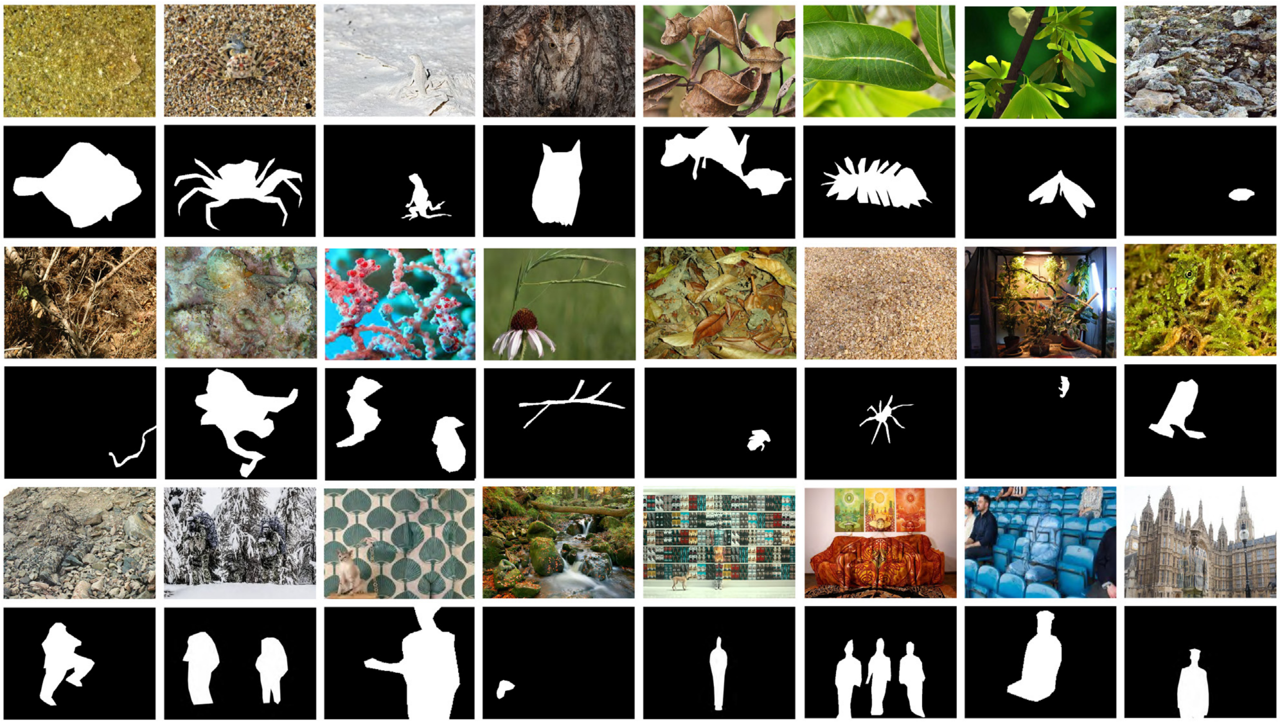


Fig. 1. A few examples from our Camouflaged Object (CAMO) dataset with corresponding pixel-level annotations. Camouflaged objects attempt to conceal their texture into the background.

As reviewed in Singh et al. (2013), there are a few methods that study camouflaged object detection in simple contexts. These methods use hand-crafted low-level features such as color, edge or texture to detect camouflaged objects. Such hand-crafted low-level features are designed to be as much discriminative as possible for detecting and segmenting objects, which is the opposite way to camouflage. Therefore, their performances are limited to camouflaged object segmentation. Furthermore, these methods can be applied only to relatively low-resolution images with a uniform background. Deep features, on the other hand, are known to outperform low-level features in many tasks in computer vision such as image classification (Krizhevsky et al., 2012), image semantic segmentation (Shelhamer et al., 2017), action recognition (Tran et al., 2015), and face recognition (Wen et al., 2016). We thus expect that deep features can replace low-level features even for camouflaged object segmentation by incorporating them into a new network. To use them, however, a large number of data of camouflaged objects are required. Nevertheless, there is no public dataset (both for training data and testing data) for the camouflaged object segmentation problem. This obviously poses serious problems to (1) train deep learning models and (2) evaluate the performance of the proposed network.

The overall contribution of this paper is two-fold:

- We provide a new image dataset of camouflaged objects to promote new methods for camouflaged object segmentation. Our newly constructed **Cam**ouflaged **O**bject (CAMO) dataset consists of 1250 images, each of which contains at least one camouflaged object. Pixel-wise ground-truths are manually annotated to each image. Furthermore, images in the CAMO dataset involve a variety of challenging scenarios such as *object appearance, background clutter, shape complexity, small object, object occlusion, multiple objects,* and *distraction*. We emphasize that this is the very first dataset for camouflage segmentation.
- We propose a novel end-to-end network, called the Anabranch[2] Network (ANet), which is *general, conceptually simple, flexible, and efficient* for camouflaged object segmentation. Our proposed ANet leverages the advantages of different network streams, namely, the classification stream and the segmentation stream, in order to segment camouflaged objects. The classification stream is used as the awareness of existing of camouflaged object(s) in an image. This design is motivated by the observation that *there is no guarantee that a camouflaged object is always present in an image* and thus we first identify whether a camouflaged object is present in an image and then only if exists, the object(s) should be accurately segmented; nothing is segmented otherwise. Extensive experiments verify the effectiveness of our proposed ANet when the network is applied on various fully convolutional networks (FCNs) for camouflaged object segmentation.

The datasets, evaluation scripts, models, and results are publicly available at our websites[3],[4].

The remainder of this paper is organized as follows. Section 2 summarizes the related work. Next, Section 3 and Section 4 introduce the constructed dataset and the proposed network, respectively. Section 5 then reports extensive experiments over the proposed method and baselines on the newly constructed dataset. Finally, Section 6 draws the conclusion and paves the way for the future work.

2. Related work

This section first reviews existing works on camouflaged object segmentation and salient object segmentation, which share some similarities. We also clarify that the camouflaged object segmentation is more challenging than the salient object segmentation. Then, we briefly introduce some advancements of two-stream networks in several computer vision tasks.

[2] An anabranch is a section of a river or stream that diverts from the main channel or stem of the watercourse and rejoins the main stem downstream.

[3] https://sites.google.com/view/ltnghia/research/camo.

[4] https://sites.google.com/site/vantam/research.

2.1. Camouflaged object segmentation

As addressed in the intensive survey compiled by Singh et al. (2013), most of the related work (Kavitha et al., 2011; Pan et al., 2011; Siricharoen et al., 2010; Song and Geng, 2010; Yin et al., 2011) uses handcrafted, low-level features for camouflaged region detection/segmentation where the similarity between the camouflaged region and the background is evaluated using low-level features, *e.g.*, color, shape, orientation, and brightness (Galun et al., 2003; Song and Geng, 2010; Xue et al., 2016). We note that there is little work that directly deals with camouflaged object detection; most of the work is dedicated to detecting the foreground region even when some of its texture is similar to the background. Pan et al. (2011) proposed a 3D convexity based method to detect camouflaged objects in images. Liu et al. (2012) integrated the top-down information based on the expectation maximization framework for foreground object detection. Sengottuvelan et al. (2008) applied a gray level co-occurrence matrix method to identify the camouflaged object in images with a simple background. In the case where some parts of a moving object and the background share the similar texture, optical flow is combined with color to detect/segment the moving object (Yin et al., 2011; Gallego and Bertolino, 2014). All these methods use handcrafted low-level features and work for only a few cases where images/videos have the simple and non-uniform background. Their performances are also unsatisfactory in camouflage detection/segmentation when there is a strong similarity between the foreground and the background.

One of the main reasons why there has been very little work on camouflaged object segmentation is the lack of the standard dataset for this problem. A benchmarking dataset is thus mandatory to develop advanced techniques for camouflaged object segmentation. Note that the dataset has to include both training data and testing data. To our best of knowledge, there is so far no standard dataset for camouflage object segmentation, which has the sufficient number of data for training and evaluating deep networks. A few of datasets related to camouflaged animals have been proposed, but they have a limited number of samples. Bideau and Learned-Miller (2016) proposed a camouflaged animal video dataset but the dataset has only 9 videos, and camouflaged animals really exist in a third of videos. The unpublished dataset proposed by Skurowski et al.[5] has only 76 images of camouflaged animals. Therefore, we collect a new camouflaged object dataset. Note that our work is the first work solve the real camouflage segmentation problem in diverse images. Different from the two above datasets, our constructed dataset consists of both camouflaged animals and human.

2.2. Salient object segmentation

Salient object predictors aim to detect and segment salient objects in images. The applications of salient object segmentation are discussed in Nguyen et al. (2018). Though "saliency" is opposed to "camouflage", techniques developed for salient object segmentation may be useful for camouflaged object segmentation. This is because the two tasks highlight image regions with certain characteristics.

Early work on salient object segmentation is based on biologically-inspired approaches (Itti et al., 1998; Koch and Ullman, 1987) where used features are the contrast of low-level features such as orientation of edges, or direction of movement. Since human vision is sensitive to color, different approaches using color were proposed where the contrast of color features is locally or globally analyzed (Achanta et al., 2009; Cheng et al., 2011; Nguyen and Sepulveda, 2015). Local analysis methods estimate the saliency of a particular image region against its neighborhoods based on color histogram comparison (Cheng et al., 2011); global analysis methods achieve globally consistent results by computing color dissimilarities to the mean image color (Achanta et al., 2009). Meanwhile, Nguyen and Sepulveda (2015) combine local and global contrast in the form of augmented hypotheses. Various patch-based methods were also proposed that estimate dissimilarity between image patches (Cheng et al., 2011; Zhang et al., 2015). To leverage the advantage of deep learning, recent methods first conduct superpixel segmentation and then feed segmented regions into a convolutional neural network (CNN) individually to obtain saliency scores (Le and Sugimoto, 2018; Li and Yu, 2015; Wang et al., 2015). More recent methods modified fully convolutional networks (FCNs) to compute a pixel-wise saliency map (Hou et al., 2017; Le and Sugimoto, 2017; Li et al., 2017a; Liu and Han, 2016; Wang et al., 2017b,a; Nguyen and Liu, 2017; Nguyen et al., 2019). Skip-layer structures are usually employed in FCNs to obtain multi-scale feature maps, which are critically needed to segment a salient object (Le and Sugimoto, 2017; Liu and Han, 2016; Hou et al., 2017). Li and Yu (2016) modified the FCN to combine pixel-wise and segment-wise feature maps for saliency value prediction. Wang et al. (2017b) integrated a pyramid pooling module and a multi-stage refinement mechanism into an FCN for salient object segmentation. Wang et al. (2017a) proposed a two-stage training method to compute saliency maps using image-level tags. Object proposal based networks, which are originally developed for category-specific object detection, are also utilized to identify salient objects (Han et al., 2018b; Le and Sugimoto, 2019). In addition, salient objects from common categories can be co-segmented by exploiting correspondence relationship among multiple relevant images (Han et al., 2018a).

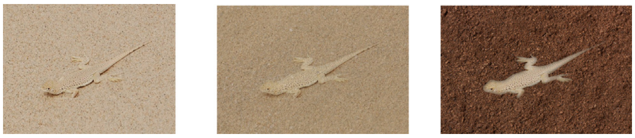

(a) Camouflaged object (b) Ambiguous (c) Salient object

Fig. 2. Ambiguous differences between camouflaged objects and salient objects.

It is worth noting that camouflaged object segmentation is more challenging than salient object segmentation. In fact, salient objects tend to be outstanding and discriminative from the background. We thus only need to focus on identifying such outstanding and discriminative regions to segment salient objects. Camouflaged objects, however, tend to conceal themselves into the background environment by decreasing discriminativeness as much as possible. This makes it hard to identify boundaries of camouflaged objects. In particular, when the background is cluttered, camouflaged objects blend with the background too much, and discriminating them from the background becomes even harder.

We remark that the same object can be either a camouflaged object or a salient object, meaning that the differences between camouflaged objects and salient objects are ambiguous (cf. Fig. 2). This suggests that some salient object detection methods may work for the camouflaged object segmentation task; however, this is not the case. Indeed, existing methods on salient object segmentation are designed to segment objects for any reason because a salient object is assumed to always exist in a training/testing image. On the contrary, there is no guarantee that a camouflaged object is always present in an image. The approach to camouflaged object segmentation should be completely different so that it allows us to handle practical scenarios where we do not know whether or not a target object exists.

2.3. Two-stream network

There are several works that use two-stream architectures (Simonyan and Zisserman, 2014a; Feichtenhofer et al., 2016; Wang et al.,

[5] http://zgwisk.aei.polsl.pl/index.php/en/research/other-research/63-animal-camouflage-analysis.

2016; Jain et al., 2017; He et al., 2017; Li et al., 2017b). Fusion networks were proposed for action recognition (Simonyan and Zisserman, 2014a; Feichtenhofer et al., 2016; Wang et al., 2016) and object segmentation (Jain et al., 2017) in videos. The network consists of an appearance stream and a motion stream, and the final result is fused from the outputs of the two streams. Two streams, however, have the same architecture for the same task despite the fact that they receive different inputs (*i.e.*, video frame and optical flow). Two streams (*i.e.*, a classification stream and a segmentation stream) of region-based networks for instance segmentation (He et al., 2017; Li et al., 2017b) have different architectures for different tasks. The two streams are built up from the region proposal network (Ren et al., 2015) and then work on specific regions.

In contrast with the above mentioned two-stream networks, the two streams (the classification stream and the segmentation stream) of our proposed ANet have different architectures mutually reinforce each other on the whole image for different tasks. The output of the classification stream is fused with that of the segmentation stream to enhance segmentation result. In this sense, the role of the classification stream is complementary to that of the segmentation stream.

3. Camouflaged object dataset

3.1. Motivation

In the literature, no dataset is available for camouflaged object segmentation. To promote camouflaged object segmentation, a publicly available dataset with pixel-wise ground-truth annotation is mandatory. Note that the dataset has to include both training data and testing data. The training data is suitable for training a deep neural network whereas the testing data is used for evaluation. Therefore, we aim to construct a dataset, **Camo**uflaged **O**bject (**CAMO**) dataset, to promote advancements in camouflaged object segmentation and its evaluation. We stress that our CAMO dataset is the very first dataset for camouflaged object segmentation.

3.2. Dataset construction

To build the dataset, we initially collected 3000 images in which at least one camouflaged object exists. We collected the images of camouflaged objects from the Internet with keywords: “camouflaged objects”, “camouflaged animals”, “concealed objects”, “hidden animals”, “camouflaged soldier”, “human body painting”, *etc*. Note that we consider both types of camouflaged objects, namely, natural and artificial camouflage as mentioned in Section 1. Then, we manually discarded images with low resolution or duplication. Finally, we ended up with 1250 images. To avoid inconsistency in labeling ground-truth, we asked three people to annotate camouflaged objects in all 1250 images individually using a custom designed interactive segmentation tool. On average, each person takes 2–5 min to annotate one image depending on its complexity. The annotation stage spanned about one month.

3.3. Dataset description

We describe in this section the Camouflaged Object (CAMO) dataset specifically designed for the task of camouflaged object segmentation. Some examples are shown in Fig. 1 with the corresponding ground-truth label annotations. We focus on two categories, *i.e.*, naturally camouflaged objects and artificially camouflaged objects, which usually correspond to animals and humans in the real world, respectively. Camouflaged animals consist of amphibians, birds, insects, mammals, reptiles, and underwater animals in various environments, *i.e.*, ground, underwater, desert, forest, mountain, and snow. Camouflaged human falls into soldiers on the battlefields and human body painting arts. The ratios for each category are shown in Fig. 3.

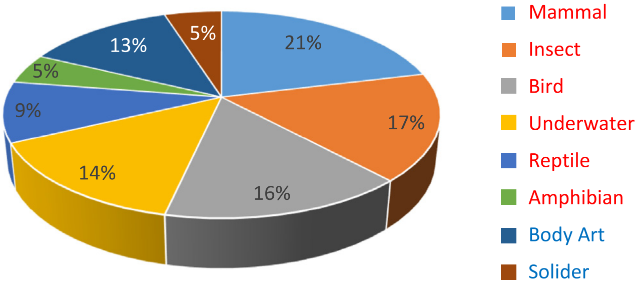
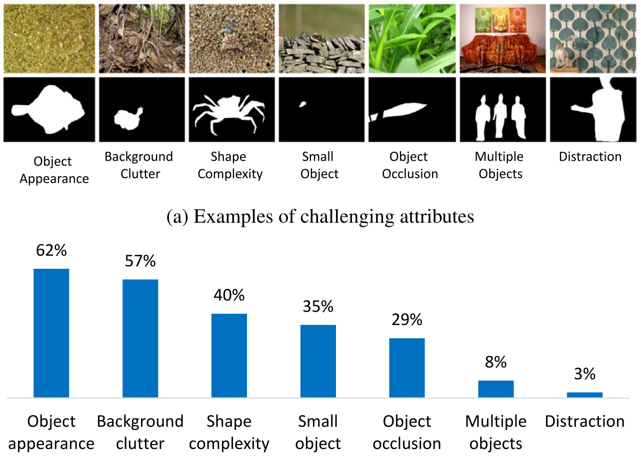


Fig. 3. Categories in Camouflaged Object Dataset. Natural camouflaged objects are shown in red while artificial camouflaged objects are shown in blue. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



(a) Examples of challenging attributes

(b) Attribute distribution over the CAMO dataset

Fig. 4. Some examples of challenging attributes and the attribute distribution over the CAMO dataset.

In our CAMO dataset, it is also noteworthy that multiple objects, including separate single objects and spatially connected/overlapping objects, possibly exist in some images (8%). This also makes our dataset more challenging for camouflaged object segmentation. The challenge of the CAMO dataset is also enhanced due to some attributes, *i.e.* object appearance, background clutter, shape complexity, small object, object occlusion, and distraction (cf. Fig. 4):

- **Object appearance**: The object has a similar color appearance with the background, causing large ambiguity in segmentation.
- **Background clutter**: The background is not uniform but contains small-scale structures or is composed of several complex parts.
- **Shape complexity**: The object has complex boundaries such as thin parts and holes, which is usually the case for the legs of insects.
- **Small object**: The ratio between the camouflaged object area and the whole image area is smaller than 0.1.
- **Object occlusion**: The object is occluded, resulting in disconnected parts or in touching the image border.
- **Distraction**: The image contains distracted objects, resulting in losing attention to camouflaged objects. This makes camouflaged objects more difficult to discover even by human beings.

4. Anabranch network

In this section, we give a detailed description of our Anabranch Network (**ANet**). Specifically, we first discuss the motivation of network design. Then, we introduce the architecture of ANet.

4.1. Network design

As mentioned above, there is no guarantee that a camouflaged object is always present in the scene. Therefore, a method that systematically segments objects for any image will not work. Moreover, directly applying discriminative features from segmentation models (*i.e.*, semantic segmentation and salient object segmentation, *etc.*) to camouflaged object segmentation is not effective because camouflaged objects conceal their texture into the surrounding environment. In order to segment camouflaged objects, we need an additional function that identifies whether a camouflaged object exists in an image. To do so, each pixel needs to be classified as a part of camouflaged objects or not. Such classification can be used not only to enhance segmentation accuracy but also to segment multiple camouflaged objects. Indeed, this classification only strengthens features extracted from the camouflaged part and weakens features extracted from a non-camouflaged part. Accordingly, segmentation and classification tasks should be closely combined in the network with different architectures for camouflaged object segmentation.

4.2. Network architecture

Fig. 5 depicts the overview of our proposed ANet, a general network for camouflaged object segmentation. The ANet leverages the advantages of the two different network models, namely, the classification network model based on a convolutional neural network (CNN) and the segmentation network model based on a fully convolutional network (FCN). The CNN reveals object hypotheses in the image (Zhou et al., 2014), yielding a cue on whether or not the input image contains camouflaged object(s). We thus aim to train the CNN model so that it classifies two classes: camouflaged image class and non-camouflaged image one. The FCN, on the other hand, provides us with the pixel-wise semantic information in the image. We consider the output of the FCN as the semantic information of different objects in the image. The two outputs from the two network streams are finally fused to produce a pixel-wisely accurate camouflage map which uniformly covers camouflaged objects. We note that our classification stream output, *i.e.*, the probability (scalar), is multiplied ($\otimes$) with each pixel value of 2D map produced by the segmentation stream.

Segmentation stream: The input and the output of the segmentation stream are both images, and, thus, any end-to-end FCN for segmentation can be employed. In the sense that we can employ any FCN for segmentation, ANet can be considered as a general network.

Though, as addressed above, methods for salient object detection alone may not work for accurate camouflaged object segmentation, effective compensation by the classification stream can be expected to boost up the performance in the ANet. Recent state-of-the-art methods for salient object detection have demonstrated their good performances. In addition, as we observed in Fig. 2, we have ambiguity in difference between camouflaged objects and salient objects. Based on these reasons, we consider a saliency model is suitable to use into ANet. We can easily swap the model with any other saliency model. We remark that it is also interesting to utilize saliency models in a similar but different domain such as camouflaged object segmentation.

Classification stream: We build the classification stream on top of convolution layers, which play as feature extraction module, of FCNs. In particular, we use three fully connected layers with 4096, 4096, and 2 filters, respectively (cf. Fig. 5 and Table 1). We note that each of the first two fully connected layers is followed by a Rectified Linear Unit (ReLU) activation layer (Krizhevsky et al., 2012). The last layer is the soft-max layer. To train this stream, we use soft-max with the cross-entropy loss. During the training process, we use dropout layers with a dropout rate of 50% after the first two fully connected layers to avoid over-fitting.

As seen above, our introduced ANet is conceptually simple. Each stream in ANet has its own task and; thus, each stream is expected to be individually trained using data suitable for the task to enhance its ability. Then, the fusion of the two streams into one boosts up segmentation accuracy of camouflaged objects. Furthermore, ANet is flexible in the sense that we can employ any end-to-end FCN and easily switch it with another. As we see in our experimental results, ANet is robust since it maintains good segmentation accuracy in both cases of with and without a camouflaged object in an input image. We also see its computational efficiency.

Table 1
Architecture of the classification stream.

No	Layer	Output	Dropping rate
1	Fully connected	2048	
2	ReLU	2048	
3	Dropout	2048	50%
4	Fully connected	2048	
5	ReLU	2048	
6	Dropout	2048	50%
7	Fully connected	2	
8	Soft-max	2	

Table 2
CAMO–COCO dataset used in our experiments.

	Training	Testing	Total
Camouflaged object images (CAMO)	1000	250	1250
Non-camouflaged object images (MS-COCO)	1000	250	1250
Total	2000	500	2500

5. Experiments

In this section, we first introduce evaluation datasets and evaluation criteria used in experiments. Then we describe implementing details as well as network training of our proposed ANet and other baselines. In experiments, we show the potential ability of transforming domain from salient object segmentation to camouflaged object segmentation. We next compare instances of our ANet with the FCN models fine-tuned on two evaluation datasets, to demonstrate that classifying camouflaged objects and non-camouflaged objects can boost up the camouflaged object segmentation. We also present the efficiency of our general ANet through short network training and fast running time. These results can be considered as the first baselines for the camouflaged object segmentation problem. Finally, we discuss the challenge of the proposed CAMO dataset to show rooms for further research on this problem.

5.1. Datasets and experimentation setup

Any image in the CAMO dataset contains at least one camouflaged object. There is thus prior information that camouflaged objects are always present. In a realistic scenario, however, there is no guarantee that a camouflaged object is always present in an image. Therefore, we set up another dataset, called CAMO–COCO, consisting of camouflaged object images and non-camouflaged object images. We used the entire images in the CAMO dataset for the camouflaged object images and collected additional 1250 images from the MS-COCO dataset (Lin et al., 2014) for the non-camouflaged object images (cf. Fig. 6). We note that we created zero-mask ground-truth labels (all pixels have zero values) for the non-camouflaged object images. For each of the camouflaged image and non-camouflaged object image sets, we randomly chose 80% for training (1000 images) and used the remaining 20% for testing (250 images). Table 2 shows the number of images in the CAMO–COCO dataset used for experiments.

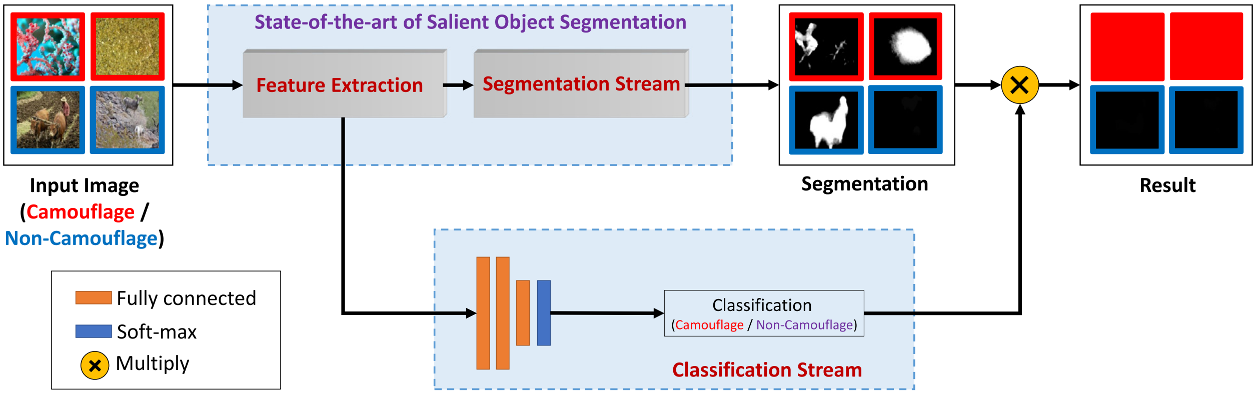


Fig. 5. Overview of our proposed Anabranch Network (ANet). The proposed network leverages the strength of both image classification and semantic segmentation tasks for camouflaged object segmentation.

Fig. 6. Examples of non-camouflaged objects from MS-COCO.

5.2. Evaluation criteria

We used the F-measure (F_β) (Achanta et al., 2009), Intersection Over Union (IOU) (Long et al., 2015), and Mean Absolute Error (MAE) as the metrics to evaluate obtained results. The first metric, F-measure, is a balanced measurement between precision and recall as follows:

$$F_\beta = \frac{\left(1+\beta^2\right) Precision \times Recall}{\beta^2 \times Precision + Recall}. \quad (1)$$

Note that we set $\beta^2 = 0.3$ as used in Achanta et al. (2009) to put an emphasis on precision. IOU is the area ratio of the overlapping against the union between the predicted camouflage map and the ground-truth map. Meanwhile, MAE is the average of the pixel-wise absolute differences between the predicted camouflage map and the ground-truth.

For MAE, we used the raw grayscale camouflage map. For the other metrics, we binarized the results depending on two contexts. In the first context, we assume that camouflaged objects are always present in every image like salient objects; we used an adaptive threshold (Jia and Han, 2013) $\theta = \mu + \eta$ where μ and η are the mean value and the standard deviation of the map, respectively. In the second context which is much closer to a real-world scenario, we assume that the existence of camouflaged objects is not guaranteed in each image; we used the fixed threshold $\theta = 0.5$.

5.3. Implementation details

To demonstrate the generality and flexibility of our proposed ANet, we employ various recent state-of-the-art FCN-based salient object segmentation models for the segmentation stream. They are DHS (Liu and Han, 2016), DSS (Hou et al., 2017), SRM (Wang et al., 2017b), and WSS (Wang et al., 2017a). For each employed model, we train two streams sequentially. We first fine-tuned the segmentation stream using the CAMO dataset from the publicly available pre-trained model and then trained the classification stream using CAMO–COCO with fixed parameters in the segmentation stream. The final model is considered as our baseline for camouflaged object segmentation.

In both training steps, we set the size of each mini-batch to 2, and used the Stochastic Gradient Descent (SGD) optimization (Rumelhart et al., 1988) with a moment $\beta = 0.9$ and a weight decay of 0.0005. We trained the segmentation stream for 10 epochs (corresponding to 5k iterations) with the learning rate of 10^{-4} and trained the classification stream for 3 epochs (3k iterations) with the learning rate of 10^{-6}. During the fine-tuning process, a simple data augmentation technique was used to avoid over-fitting. Images were rescaled to the resolution of the specific FCN employed in the segmentation stream and randomly flipped horizontally. We note that for comparison, our employed FCN models (original) were trained on CAMO (or CAMO–COCO) dataset for 10 epochs with the learning rate of 10^{-4} and the other parameters were set similarly to our ANet. We also remark that we implemented our method in C/C++, using Caffe (Jia et al., 2014) toolbox and conducted all the experiments on a computer with a Core i7 3.6 GHz processor, 32 GB of RAM, and two GTX 1080 GPUs.

5.4. Experimental results

5.4.1. Baseline evaluation

We evaluated our model (baseline) on two datasets: CAMO (camouflaged object images only) and CAMO–COCO (both camouflaged and non-camouflaged object images) where only test images in each dataset were used for this evaluation. We employed an FCN (DHS (Liu and Han, 2016), DSS (Hou et al., 2017), SRM (Wang et al., 2017b), or WSS (Wang et al., 2017a)) as the segmentation stream. For comparison, we also evaluated each of used FCNs to see how it works for camouflaged object segmentation. Each was evaluated with three different models: the pre-trained models on saliency datasets (*e.g.* MSRA (Achanta et al., 2009), DUT-OMRON (Yang et al., 2013), and DUTS (Wang et al., 2017a)), the models that were fine-tuned with CAMO or CAMO–COCO (the model is specified by the word inside the parentheses). Figs. 7, 8, and Table 3 illustrate the obtained results of our experiments. We note that ANet-DHS (baseline), for example, denotes our proposed model employing DHS as the segmentation stream.

Figs. 7 and 8 show that ANet yields better results than its employed original FCN (pre-trained) independently of the choice of the FCN. In particular, for non-camouflaged object images, the superiority of performances of ANet against its original FCN is distinguished. This suggests that the fusion with the classification stream indeed works effectively to boost the segmentation accuracy.

Table 3
Experimental results on two datasets: CAMO dataset (the left part), and CAMO–COCO dataset (the right part). The evaluation is based on F-measure (Achanta et al., 2009) (the higher the better), IOU (Long et al., 2015) (the higher the better), and MAE (the smaller the better). The 1st and 2nd places are shown in **blue** and **red**, respectively.

Dataset in test	**CAMO**					**CAMO-COCO**				
Method		**Adaptive Threshold**		**Fixed Threshold**			**Adaptive Threshold**		**Fixed Threshold**	
	MAE ⇓	**F**$_\beta$ ⇑	**IOU** ⇑	**F**$_\beta$ ⇑	**IOU** ⇑	**MAE** ⇓	**F**$_\beta$ ⇑	**IOU** ⇑	**F**$_\beta$ ⇑	**IOU** ⇑
DHS (Liu and Han, 2016) (pre-trained)	0.173	0.548	0.351	0.562	0.332	0.204	0.746	0.571	0.750	0.549
DHS (fine-tuned with CAMO)	**0.129**	**0.640**	**0.459**	**0.643**	**0.444**	**0.169**	0.794	0.633	0.793	0.618
DHS (fine-tuned with CAMO-COCO)	0.138	0.596	0.388	0.614	0.367	**0.072**	**0.796**	**0.679**	**0.808**	**0.681**
ANet-DHS (baseline)	**0.130**	**0.626**	**0.437**	**0.631**	**0.423**	**0.072**	**0.812**	**0.712**	**0.814**	**0.705**
DSS (Hou et al., 2017) (pre-trained)	0.157	0.564	0.320	0.563	0.333	0.176	0.757	0.559	0.756	0.570
DSS (fine-tuned with CAMO)	**0.141**	**0.622**	**0.425**	**0.631**	**0.420**	0.152	**0.791**	0.633	**0.795**	0.630
DSS (fine-tuned with CAMO-COCO)	0.145	0.582	0.385	0.584	0.381	**0.076**	0.790	**0.686**	0.792	**0.687**
ANet-DSS (baseline)	**0.132**	**0.587**	**0.404**	**0.607**	**0.390**	**0.067**	**0.795**	**0.701**	**0.804**	**0.694**
SRM (Wang et al., 2017b) (pre-trained)	0.171	0.448	0.258	0.425	0.213	0.191	0.699	0.535	0.685	0.502
SRM (fine-tuned with CAMO)	**0.120**	**0.683**	**0.507**	**0.688**	**0.498**	0.176	0.815	0.651	0.812	0.634
SRM (fine-tuned with CAMO-COCO)	0.127	**0.663**	0.454	0.656	0.421	**0.067**	**0.830**	**0.717**	**0.831**	**0.708**
ANet-SRM (baseline)	**0.126**	0.654	**0.475**	**0.662**	**0.466**	**0.069**	**0.826**	**0.732**	**0.830**	**0.727**
WSS (Wang et al., 2017a) (pre-trained)	0.178	0.559	0.323	0.531	0.265	0.197	0.754	0.567	0.740	0.528
WSS (fine-tuned with CAMO)	**0.145**	**0.658**	**0.477**	**0.661**	**0.465**	0.174	0.807	0.649	**0.810**	0.637
WSS (fine-tuned with CAMO-COCO)	0.149	0.642	0.439	0.638	0.382	**0.085**	**0.811**	**0.678**	**0.820**	**0.687**
ANet-WSS (baseline)	**0.140**	**0.661**	**0.459**	**0.643**	**0.407**	**0.078**	**0.826**	**0.710**	**0.820**	**0.697**

5.4.2. *Performance of Camouflaged object segmentation*

It is interesting for the saliency community to see the segmentation capabilities of saliency systems to segment camouflaged objects (which should actually not be possible due to different properties of those objects). Therefore, we show results of FCNs using pre-trained models for salient object segmentation in Table 3. Although results are moderate (the worst results among compared methods for each FCN), it also illustrates the potential ability to transfer from salient object segmentation to camouflaged object segmentation due to the ambiguous border of these kinds of object. This shows the possibility of using existing saliency models to tackle camouflaged object segmentation problem.

Table 3 shows that the fine-tuned FCN models achieve around 60+ % accuracy in F_β and IOU (around 0.1 in MAE). Naturally, the model fine-tuned with the dataset used for testing performs better than the pre-trained model or the model fine-tuned with the other dataset. We also observe the model fine-tuned with CAMO–COCO performs better than the one fine-tuned with CAMO. This is because CAMO is a subset of CAMO–COCO. These mean that salient object segmentation methods are, to some extent, capable of segmenting even a not-target object such as a camouflaged object. This may come from the ambiguity of the differences between salient objects and camouflaged objects. This, simultaneously, also supports the use of an FCN developed for salient object segmentation as the segmentation stream in our ANet. We then draw our attention to the performances of ANet using an FCN. Since the segmentation stream of ANet is trained using CAMO and its output is multiplied by the output of the classification stream, the segmentation ability of ANet is limited by that of the employed FCN fine-tuned with CAMO. In other words, the performance of ANet on CAMO is expected to be comparable with that of the employed FCN (fine-tuned with CAMO), which is confirmed in Table 3. When we evaluate ANet on CAMO–COCO, however, the performance of ANet can be better than that of the employed FCN (fine-tuned with CAMO). This can be explained by the nature of CAMO–COCO where an image may not contain any camouflaged object, and thus the role of the classification stream becomes more crucial; in other words, the accuracy of the classification stream boosts up the segmentation accuracy of ANet. The segmentation accuracy of the FCN, in contrast, is degraded because it is not trained on CAMO–COCO. A similar conclusion can be derived from the comparison between ANet and the employed FCN fine-tuned with CAMO–COCO. Accordingly, ANet works with comparable accuracy on *both* CAMO and CAMO–COCO while each FCN works well *only on one* of the two datasets. Indeed, this can be observed in Table 3. This clearly demonstrates that ANet can maintain with flexibility a good segmentation accuracy, regardless whether a camouflaged object is present in the input image or not.

We also statistically evaluated the significance of the differences using the t-test with the significance level of 90%. For CAMO–COCO dataset, we confirmed that on all the metrics, ANet significantly outperforms the other methods mostly or is at least similar to the best methods. For CAMO dataset, we also confirmed ANet is significantly better than FCNs fine-tuned on CAMO–COCO, and FCNs fine-tuned on CAMO are significantly better than ANet. This means ANet's ability for CAMO dataset is between FCNs fine-tuned on CAMO and FCNs fine-tune on CAMO–COCO. This is reasonable because the performance of the classification stream is not perfect, leading to slightly reducing the performance on CAMO. To summarize, ANet has good performance for *both* CAMO and CAMO–COCO datasets. On the other hand, FCNs fine-tuned with CAMO–COCO have good performance for only CAMO–COCO but not for CAMO; whereas FCNs fine-tuned with CAMO have good performance for only CAMO but not for CAMO–COCO (*not for both*).

5.4.3. *Performance of Camouflaged object classification*

In order to evaluate the level of accuracy achieved by our classification stream, we compared the performance of our classification stream with that of the SVM-based classification using the Bag-of-Words

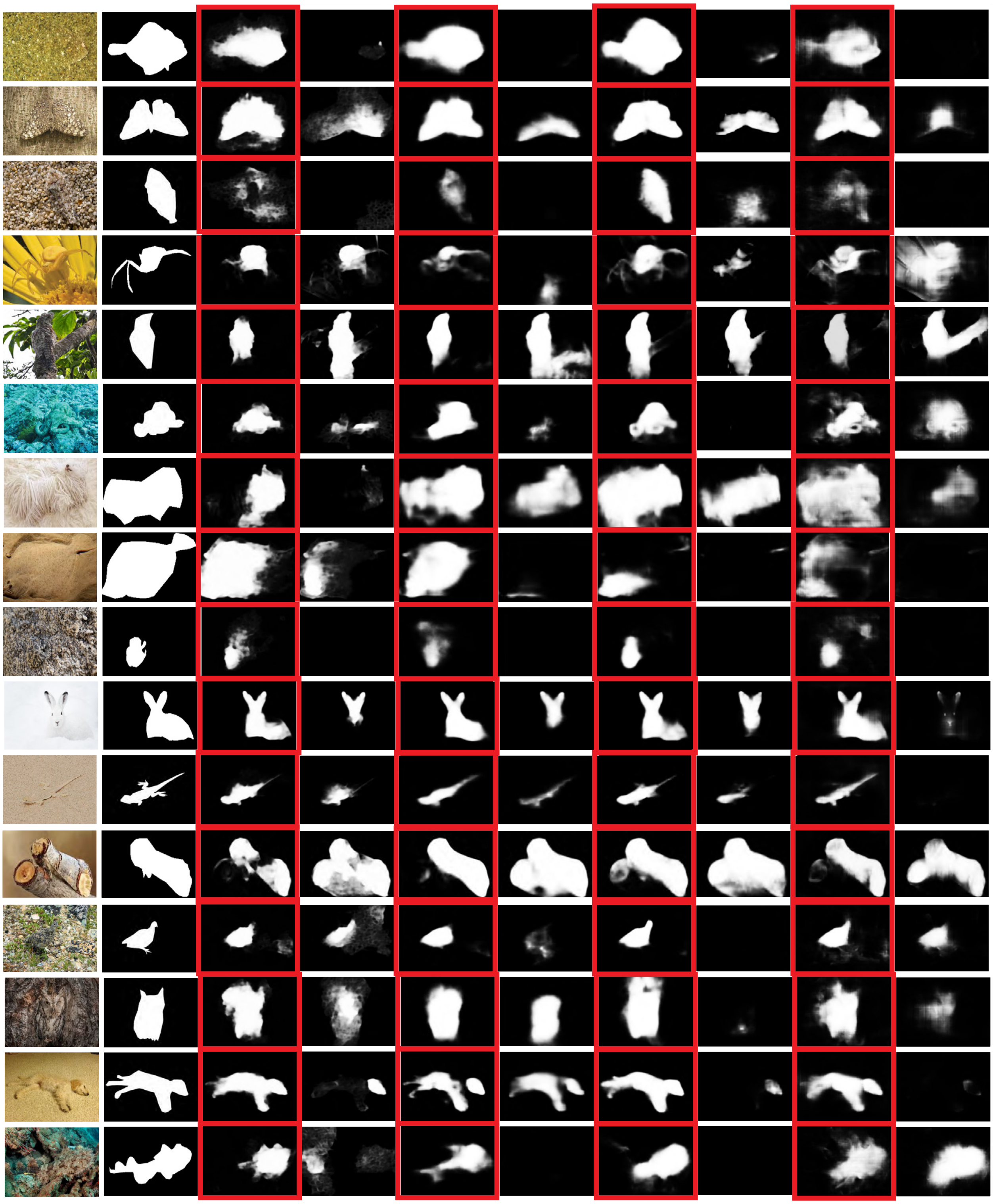


Fig. 7. Visual comparison of our ANet employing different FCNs on natural camouflaged object images (CAMO dataset). From left to right, input image and ground-truth are followed by outputs obtained using ANet with DHS (Liu and Han, 2016), DHS (pre-trained), ANet with DSS (Hou et al., 2017), DSS (pre-trained), ANet with SRM (Wang et al., 2017b), SRM (pre-trained), ANet with WSS (Wang et al., 2017a), and WSS (pre-trained) in this order. The results obtained by ANet is surrounded with red rectangles. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

model (Fei-Fei and Perona, 2005) with SIFT features (Lowe, 1999) (denoted by SVM-BoW) and CNNs, including AlexNet (Krizhevsky et al., 2012) and VGG-16 (Simonyan and Zisserman, 2014b). For CNNs, we fine-tuned them on our CAMO–COCO dataset from pre-trained models on the ImageNet dataset (Russakovsky et al., 2015). The results are shown in Table 4, indicating that the accuracy of our classification stream achieves around 90%, which outperforms SVM-BoW. ANet baseline models are also significantly better than both AlexNet and VGG-16. As a closer look, the classification stream achieves high accuracy on few training iterations (3 epochs).

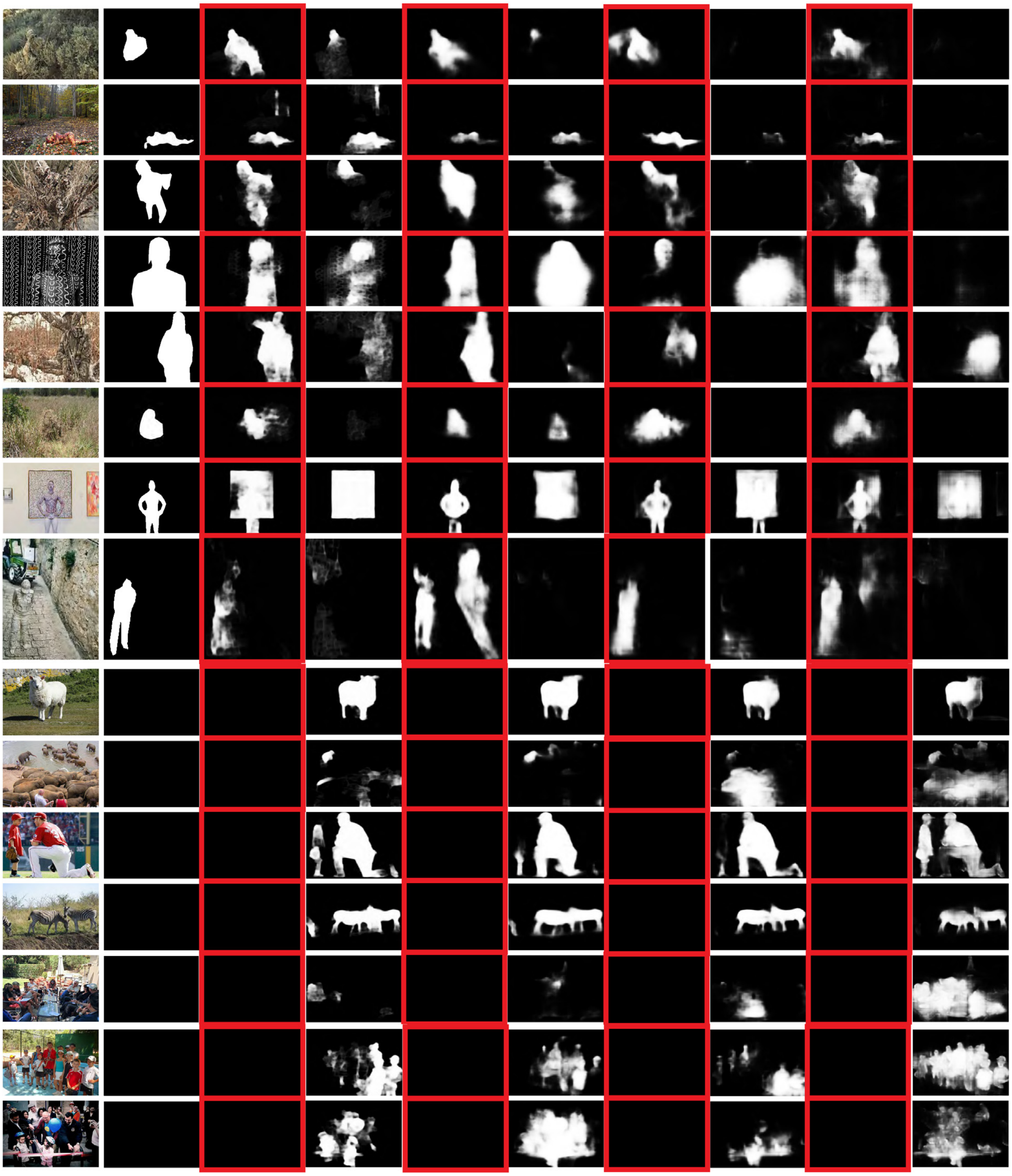


Fig. 8. Visual comparison of our ANet employing different FCNs. From left to right, input image and ground-truth are followed by outputs obtained using ANet with DHS (Liu and Han, 2016), DHS (pre-trained), ANet with DSS (Hou et al., 2017), DSS (pre-trained), ANet with SRM (Wang et al., 2017b), SRM (pre-trained), ANet with WSS (Wang et al., 2017a), and WSS (pre-trained) in this order. The first eight rows are images with artificial camouflaged objects (CAMO dataset), and the last seven rows are images without camouflaged objects (COCO dataset). The results obtained by ANet is surrounded with red rectangles. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.4.4. Computational efficiency

We further evaluated the computational efficiency of all the baseline models. We also evaluated the average running time of the original FCN models employed in our ANet. The results are shown in Table 5. We observe that stacking the classification stream does not incur much more processing time, indicating that the ANet is able to keep the running time computationally efficient.

5.5. Discussion

5.5.1. Failure cases

The accuracy of our baselines on CAMO for camouflaged object segmentation is far lower than the state-of-the-art accuracy for salient object segmentation, which can achieve up to round 90% in F_β. This is mainly caused by the insufficient ability of the segmentation stream

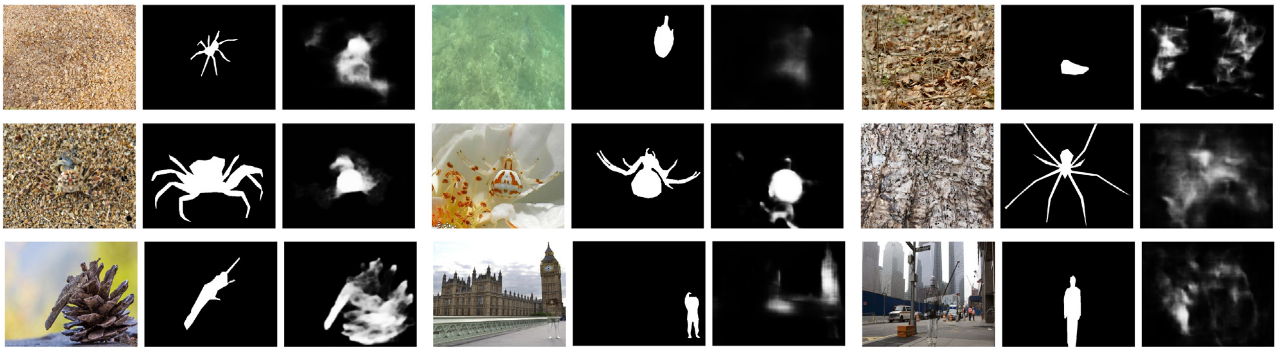


Fig. 9. Failure cases of FCNs on CAMO. Each case is shown in a horizontal triplet of images, from left to right: input image, ground-truth, and result by the best FCN (fine-tuned with CAMO) in this order.

Table 4
The accuracy (%) of camouflaged object classification on the CAMO–COCO dataset.

Method	Accuracy
SVM-BoW (Fei-Fei and Perona, 2005)	54.0
AlexNet (Krizhevsky et al., 2012)	76.0
VGG-16 (Simonyan and Zisserman, 2014b)	88.6
ANet-DHS	91.4
ANet-DSS	89.2
ANet-SRM	90.6
ANet-WSS	89.6

Table 5
The average wall-clock time in millisecond for each image.

Method	FCN	ANet (baseline)
DHS (Liu and Han, 2016)	63	73
DSS (Hou et al., 2017)	76	79
SRM (Wang et al., 2017b)	81	86
WSS (Wang et al., 2017a)	61	65

when we transfer domain from salient object segmentation to camouflaged object segmentation. Fig. 9 shows some failure segmentation results obtained by our used FCNs where input images involve challenging scenarios in CAMO (object appearance and background clutter in the first row; the shape complexity of insect legs in the second row; distraction in the third row).

5.5.2. Joint training

The classification stream is expected to boost the segmentation accuracy in ANet, however, fusing the two streams by the multiplication is still insufficient. Further effective fusion of the two streams should be explored to improve the segmentation accuracy.

For a two-stream network, jointly training two streams after training each stream separately is known as a strategy to improve the performance (Jain et al., 2017). Following this strategy, we jointly trained the ANet (baseline) using CAMO–COCO where we used 10 epochs (10k iterations) with the learning-rates of 10^{-4} (for the segmentation stream) and 10^{-6} (for the classification stream). Other parameters are set similarly to training two streams individually. The loss function for joint-training is defined as the summation of the loss of the classification stream and that of the segmentation stream:

$$\mathcal{L} = \mathcal{L}_{\text{seg}}(\hat{y}_{\text{seg}}, y_{\text{seg}}) + \mathcal{L}_{\text{cls}}(\hat{y}_{\text{cls}}, y_{\text{cls}}), \tag{2}$$

where $\hat{y}_{\text{seg}}$, y_{seg}, $\hat{y}_{\text{cls}}$, and y_{cls} denote outputs and ground-truth labels of the segmentation stream and the classification stream, respectively. $\mathcal{L}_{\text{cls}}$ is the soft-max loss of the classification stream and $\mathcal{L}_{\text{seg}}$ is the specific loss of the corresponding FCN implemented in the segmentation stream.

Table 6 illustrates the improvement of performances by introducing the joint-training to ANet. We see that differently from other two-stream network cases, jointly-training ANet does not lead to better performance for all methods. DHS and WSS reduce accuracy while DSS and SRM slightly increase accuracy (see Table 6). This can be explained as follows. Each stream of ANet has its own task and thus using training data suitable for the task is mandatory to enhance its ability. During the joint-training of ANet, however, data used for training cannot be suitable for each of the tasks of the two streams, resulting in interfering with each other stream. This is also facilitated by the branch structure of ANet. Accordingly, joint-training potentially brings to ANet some collapse in improving training of each stream.

Therefore, jointly training two streams may not bring gain to ANet due to opposite attributes of camouflaged objects and non-camouflaged objects. The two-step training approach is sufficient for ANet for the task of camouflaged object segmentation on the CAMO–COCO dataset. Further investigation on this issue is left for future work.

6. Conclusion

In this paper, we addressed an interesting yet challenging problem of camouflaged object segmentation by providing a new image dataset of camouflaged object segmentation where each image is manually annotated with pixel-wise ground-truth. We believe that our novel dataset will promote new advancements on camouflaged object segmentation. We also aim to explore camouflaged object segmentation on videos in the near future.

We proposed a simple and flexible end-to-end network, namely Anabranch Network, for camouflaged object segmentation where the classification stream and the segmentation stream are effectively combined to show the baseline performance. To show effectiveness of our proposed framework, which can boot the segmentation using classification, we applied it to different FCNs. Extensive experiments conducted on the newly built dataset demonstrate the superiority of the proposed network. In addition, our method is computationally efficient.

This paper focused on only regions, but the potential idea of utilizing classification into segmentation can be useful when applying for instance segmentation in appropriate ways. For examples, instance classification can be combined with instance segmentation, where each instance is classified whether it is camouflaged or salient. This is left for the future work.

Acknowledgments

This work is in part funded by University of Dayton SEED Grant (United States), and by SOKENDAI Short-Stay Study Abroad Program (Japan). We also thank Vamshi Krishna Madaram and Zhe Huang (University of Dayton) for their support in the annotation of the CAMO dataset. We gratefully acknowledge NVIDIA for the support of GPU. The authors declare no conflict of interest.

Table 6
Performance of joint-training on CAMO dataset and CAMO–COCO dataset. The best results are shown in **blue**.

Dataset in test	CAMO					CAMO-COCO				
Method		Adaptive Threshold		Fixed Threshold			Adaptive Threshold		Fixed Threshold	
	MAE ⇓	F_β ⇑	IOU ⇑	F_β ⇑	IOU ⇑	MAE ⇓	F_β ⇑	IOU ⇑	F_β ⇑	IOU ⇑
ANet-DHS (baseline)	0.130	0.626	0.437	0.631	0.423	0.072	0.812	0.712	0.814	0.705
ANet-DHS (+joint training)	0.201	0.509	0.314	0.490	0.130	0.109	0.749	0.628	0.758	0.564
ANet-DSS (baseline)	0.132	0.587	0.404	0.607	0.390	0.067	0.795	0.701	0.804	0.694
ANet-DSS (+joint training)	0.126	0.644	0.441	0.651	0.417	0.064	0.822	0.719	0.825	0.707
ANet-SRM (baseline)	0.126	0.654	0.475	0.662	0.466	0.069	0.826	0.732	0.830	0.727
ANet-SRM (+joint training)	0.123	0.680	0.481	0.682	0.454	0.063	0.839	0.733	0.841	0.726
ANet-WSS (baseline)	0.140	0.661	0.459	0.643	0.407	0.078	0.826	0.710	0.820	0.697
ANet-WSS (+joint training)	0.145	0.626	0.394	0.628	0.324	0.074	0.813	0.692	0.816	0.662

References

Achanta, R., Hemami, S., Estrada, F., Susstrunk, S., 2009. Frequency-tuned salient region detection. In: Conference on Computer Vision and Pattern Recognition. pp. 1597–1604.

Bideau, P., Learned-Miller, E., 2016. It's moving! a probabilistic model for causal motion segmentation in moving camera videos. In: European Conference on Computer Vision.

Cheng, M.M., Zhang, G.X., Mitra, N., Huang, X., Hu, S.M., 2011. Global contrast based salient region detection. In: Conference on Computer Vision and Pattern Recognition. pp. 409–416.

Fei-Fei, L., Perona, P., 2005. A bayesian hierarchical model for learning natural scene categories. In: Conference on Computer Vision and Pattern Recognition, Vol. 2. pp. 524–531.

Feichtenhofer, C., Pinz, A., Wildes, R., 2016. Spatiotemporal residual networks for video action recognition. In: Advances in Neural Information Processing Systems. pp. 3468–3476.

Gallego, J., Bertolino, P., 2014. Foreground object segmentation for moving camera sequences based on foreground-background probabilistic models and prior probability maps. In: International Conference on Image Processing. pp. 3312–3316.

Galun, M., Sharon, E., Basri, R., Brandt, A., 2003. Texture segmentation by multiscale aggregation of filter responses and shape elements. In: International Conference on Computer Vision. pp. 716–723.

Han, J., Quan, R., Zhang, D., Nie, F., 2018a. Robust object co-segmentation using background prior. IEEE Trans. Image Process. 27, 1639–1651.

Han, J., Zhang, D., Cheng, G., Liu, N., Xu, D., 2018b. Advanced deep-learning techniques for salient and category-specific object detection: A survey. IEEE Signal Process. Mag. 35, 84–100.

He, K., Gkioxari, G., Dollár, P., Girshick, R., 2017. Mask r-cnn. In: International Conference on Computer Vision. pp. 2980–2988.

Hou, Q., Cheng, M.M., Hu, X., Borji, A., Tu, Z., Torr, P., 2017. Deeply supervised salient object detection with short connections. In: Conference on Computer Vision and Pattern Recognition.

Itti, L., Koch, C., Niebur, E., 1998. A model of saliency-based visual attention for rapid scene analysis. IEEE Trans. Pattern Anal. Mach. Intell. 20, 1254–1259.

Jain, S.D., Xiong, B., Grauman, K., 2017. Fusionseg: Learning to combine motion and appearance for fully automatic segmention of generic objects in videos. In: Conference on Computer Vision and Pattern Recognition.

Jia, Y., Han, M., 2013. CaTegory-independent object-level saliency detection. In: International Conference on Computer Vision. pp. 1761–1768.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T., 2014. CaFfe: Convolutional architecture for fast feature embedding. In: International Conference on Multimedia. pp. 675–678.

Kavitha, C., Rao, B.P., Govardhan, A., 2011. An efficient content based image retrieval using color and texture of image subblocks. Int. J. Eng. Sci. Technol. 3.

Koch, C., Ullman, S., 1987. Shifts in selective visual attention: Towards the underlying neural circuitry. In: Matters of Intelligence: Conceptual Structures in Cognitive Neuroscience. pp. 115–141.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems. pp. 1097–1105.

Le, T.N., Sugimoto, A., 2017. Deeply supervised 3d recurrent fcn for salient object detection in videos. In: British Machine Vision Conference.

Le, T., Sugimoto, A., 2018. Video salient object detection using spatiotemporal deep features. IEEE Trans. Image Process. 27, 5002–5015.

Le, T., Sugimoto, A., 2019. Semantic instance meets salient object: Study on video semantic salient instance segmentation. In: IEEE Winter Conference on Applications of Computer Vision. pp. 1779–1788.

Li, Y., Qi, H., Dai, J., Ji, X., Wei, Y., 2017b. Fully convolutional instance-aware semantic segmentation. In: Conference on Computer Vision and Pattern Recognition.

Li, G., Xie, Y., Lin, L., Yu, Y., 2017a. Instance-level salient object segmentation. In: Conference on Computer Vision and Pattern Recognition.

Li, G., Yu, Y., 2015. Visual saliency based on multiscale deep features. In: Conference on Computer Vision and Pattern Recognition. pp. 5455–5463.

Li, G., Yu, Y., 2016. Deep contrast learning for salient object detection. In: Conference on Computer Vision and Pattern Recognition. pp. 478–487.

Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L., 2014. Microsoft coco: Common objects in context. In: European Conference on Computer Vision. pp. 740–755.

Liu, N., Han, J., 2016. Dhsnet: Deep hierarchical saliency network for salient object detection. In: Conference on Computer Vision and Pattern Recognition. pp. 678–686.

Liu, Z., Huang, K., Tan, T., 2012. Foreground object detection using top-down information based on em framework. IEEE Trans. Image Process. 21, 4204–4217.

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In: Conference on Computer Vision and Pattern Recognition. pp. 3431–3440.

Lowe, D.G., 1999. Object recognition from local scale-invariant features. In: International Conference on Computer Vision, Pp. pp. 1150–1157.

Nguyen, T.V., Liu, L., 2017. Salient object detection with semantic priors. In: International Joint Conference on Artificial Intelligence. pp. 4499–4505.

Nguyen, T.V., Nguyen, K., Do, T.T., 2019. Semantic prior analysis for salient object detection. IEEE Trans. Image Process..

Nguyen, T.V., Sepulveda, J., 2015. Salient object detection via augmented hypotheses. In: International Joint Conference on Artificial Intelligence. pp. 2176–2182.

Nguyen, T.V., Zhao, Q., Yan, S., 2018. Attentive systems: A survey. Int. J. Comput. Vis. 126, 86–110.

Pan, Y., Chen, Y., Fu, Q., Zhang, P., Xu, X., 2011. Study on the camouflaged target detection method based on 3d convexity. Mod. Appl. Sci. 5 (152).

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems. pp. 91–99.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1988. Learning representations by back-propagating errors. In: Neurocomputing: Foundations of Research. pp. 696–699.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L., 2015. Imagenet large scale visual recognition challenge. Int. J. Comput. Vis. 115, 211–252.

Sengottuvelan, P., Wahi, A., Shanmugam, A., 2008. Performance of decamouflaging through exploratory image analysis. In: International Conference on Emerging Trends in Engineering and Technology. pp. 6–10.

Shelhamer, E., Long, J., Darrell, T., 2017. Fully convolutional networks for semantic segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 39, 640–651.

Simonyan, K., Zisserman, A., 2014a. Two-stream convolutional networks for action recognition in videos. In: Advances in Neural Information Processing Systems. pp. 568–576.

Simonyan, K., Zisserman, A., 2014b. Very deep convolutional networks for large-scale image recognition. ILSVRC.

Singh, S., Dhawale, C., Misra, S., 2013. Survey of object detection methods in camouflaged image. IERI Procedia 4, 351–357.

Siricharoen, P., Aramvith, S., Chalidabhongse, T.H., Siddhichai, S., 2010. Robust outdoor human segmentation based on color-based statistical approach and edge combination. In: International Conference on Green Circuits and Systems. pp. 463–468.

Song, L., Geng, W., 2010. A new camouflage texture evaluation method based on wssim and nature image features. In: International Conference on Multimedia Technology. pp. 1–4.

Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M., 2015. Learning spatiotemporal features with 3D convolutional networks. In: International Conference on Computer Vision. pp. 4489–4497.

Wang, T., Borji, A., Zhang, L., Zhang, P., Lu, H., 2017b. A stagewise refinement model for detecting salient objects in images. In: International Conference on Computer Vision.

Wang, L., Lu, H., Ruan, X., Yang, M.H., 2015. Deep networks for saliency detection via local estimation and global search. In: Conference on Computer Vision and Pattern Recognition. pp. 3183–3192.

Wang, L., Lu, H., Wang, Y., Feng, M., Wang, D., Yin, B., Ruan, X., 2017a. Learning to detect salient objects with image-level supervision. In: Conference on Computer Vision and Pattern Recognition.

Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Val Gool, L., 2016. Temporal segment networks: Towards good practices for deep action recognition. In: European Conference on Computer Vision.

Wen, Y., Zhang, K., Li, Z., Qiao, Y., 2016. A discriminative feature learning approach for deep face recognition. In: European Conference on Computer Vision. pp. 499–515.

Xue, F., Yong, C., Xu, S., Dong, H., Luo, Y., Jia, W., 2016. CaMouflage performance analysis and evaluation framework based on features fusion. Multimedia Tools Appl. 75, 4065–4082.

Yang, C., Zhang, L., Lu, H., Ruan, X., Yang, M.H., 2013. Saliency detection via graph-based manifold ranking. In: Conference on Computer Vision and Pattern Recognition. pp. 3166–3173.

Yin, J., Han, Y., Hou, W., Li, J., 2011. Detection of the mobile object with camouflage color under dynamic background based on optical flow. Procedia Eng. 15, 2201–2205.

Zhang, J., Sclaroff, S., Lin, Z., Shen, X., Price, B., Mech, R., 2015. Minimum barrier salient object detection at 80 fps. In: International Conference on Computer Vision. pp. 1404–1412.

Zhou, B., Khosla, A., Lapedriza, À., Oliva, A., Torralba, A., 2014. Object detectors emerge in deep scene cnns. In: International Conference on Learning Representations.