This WACV 2022 paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Modeling Aleatoric Uncertainty for Camouflaged Object Detection

Jiawei Liu, Jing Zhang, Nick Barnes
Australian National University
{jiawei.liu3, jing.zhang, nick.barnes}@anu.edu.au

Abstract

Aleatoric uncertainty captures noise within the observations. For camouflaged object detection, due to similar appearance of the camouflaged foreground and the background, it's difficult to obtain highly accurate annotations, especially annotations around object boundaries. We argue that training directly with the "noisy" camouflage map may lead to a model of poor generalization ability. In this paper, we introduce an explicitly aleatoric uncertainty estimation technique to represent predictive uncertainty due to noisy labeling. Specifically, we present a confidence-aware camouflaged object detection (COD) framework using dynamic supervision to produce both an accurate camouflage map and a reliable "aleatoric uncertainty". Different from existing techniques that produce deterministic prediction following the point estimation pipeline, our framework formalises aleatoric uncertainty as probability distribution over model output and the input image. We claim that, once trained, our confidence estimation network can evaluate the pixel-wise accuracy of the prediction without relying on the ground truth camouflage map. Extensive results illustrate the superior performance of the proposed model in explaining the camouflage prediction. Our codes are available at `https://github.com/Carlisle-Liu/OCENet`

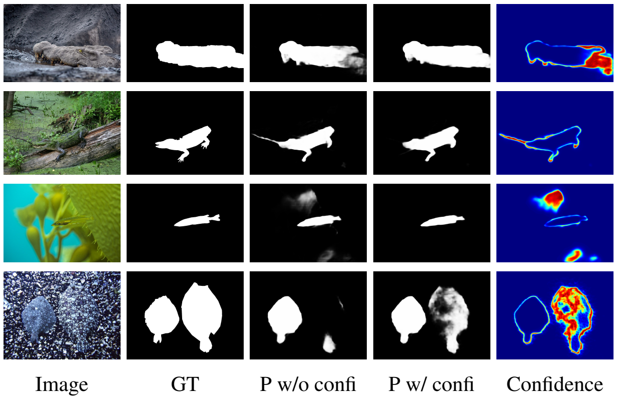

Figure 1. Prediction of our confidence-aware COD network. Red indicates low confidence and blue indicates high confidence. From left to right: input image, ground truth, prediction without confidence as guidance, and with confidence as guidance, and confidence map. For easier samples in the first and second rows, the low-confidence regions mainly distribute along object boundaries. For hard samples in the third and fourth rows, our confidence map can effectively identify the false-positive regions (third row) leading to their removal and true-negative regions (fourth row) greatly improving coverage of the second object which is difficult to see.

1. Introduction

Deep learning systems have found popularity in real-world applications, *e.g*. autonomous driving. However, failures of such Deep Neural Network (DNN) models can lead to catastrophic consequences, raising questions on their reliability. Thus, it is critical to be able to interpret the DNN model predictions in terms of uncertainty. Conventionally, two main types of uncertainties [22] exist in exiting deep nerual networks, namely *aleatoric* uncertainty representing the noise inherent in the data distribution, *e.g*. annotation ambiguity and *epistemic* uncertainty that captures the uncertainty in the model prediction. Epistemic uncertainty can be reduced by having enough data observations.

A good deal of research has been done to model the two types of uncertainties. They usually adopt the Bayesian Neural Network (BNN) framework [37, 48, 34, 17, 5, 49, 32, 3, 16, 25, 53, 1]. The main issue of BNN for uncertainty estimation is the intractable posterior inference, and so most existing uncertainty estimation techniques focus on designing approximate posterior inference. Among them, [39] learns a discriminator to distinguish between prediction and ground truth, where the output of the discriminator is defined as a confidence map or inverse uncertainty map. [36, 8] optimise the confidence estimation network with a ranking loss that assigns higher uncertainties to wrongly predicted samples or pixels. Specifically, [36] develops a correctness ranking loss that enforces that samples with higher accuracy have higher confidence, and [8] proposes a loss function that maximises the difference between the estimated confidences of correct predictions and wrong predictions. [46] presents a multi-task learning loss function derived by maximising the Gaussian likelihood with respect to the noise parameters representing Homoscedastic uncertainties, which is a type of aleatoric uncertainty that is independent of the input image.

Camouflage is defined as a state where the object has disguised appearance that is indiscernible from its surroundings, which is a widely applied technique in the world of animals to conceal themselves, deceiving predators into making false judgements. This is achieved through various camouflage techniques, *e.g.* disruptive colouration, self-decoration, cryptic behaviour, *etc*. [2, 6, 15]. This natural occurrence also inspires the development of artificial camouflage, such as military camouflage [2]. The indistinguishability of camouflaged animal poses a great challenge to annotation which becomes more prone to noisy labels. We propose to capture such annotation inconsistency by modelling the aleatoric uncertainty.

The existing techniques [22, 40] for aleatoric uncertainty estimation involve an extra variance estimation module to represent the aleatoric uncertainty. The unbounded variance is maximised at wrong predictions in order to minimise the loss, with an L2 regularisation employed to prevent it from becoming infinitely large. Differently, we propose an innovative Online Confidence Estimation Network (OCENet) to model the aleatoric uncertainty in the camouflaged object detection. We dynamically derive the difference between prediction and ground truth as supervision for the uncertainty estimation module within our OCENet. With this setting, our OCENet is able to identify wrongly classified areas as uncertain and assign low uncertainty values to correctly predicted areas. As shown in Fig. 1, our estimated confidence map is able to assign high uncertainty to under-segmentation, over-segmentation, phantom segmentation where false foreground predictions are distant from the target object, and object boundaries where errors are prone to occur.

We summarise our main contributions as: 1) We propose an innovative Online Confidence Estimation Network (OCENet) to model the aleatoric uncertainty for camouflaged object detection. It outputs pixel-wise uncertainty revealing both true-negative and false-positive predictions to prevent the network becoming overconfident; 2) Our OCENet provides an initial evaluation of the prediction without relying on the ground truth; 3) We further present a difficulty-aware learning camouflaged object detection framework to effectively utilizing the aleatoric uncertainty for hard-negative mining. Experimental results show superior performance of our model in explaining the model prediction.

2. Related Work

Confidence estimation has become an active research field in deep neural network based tasks, which is usually related to uncertainty estimation [22, 54] that models the uncertainty of model predictions. Two main uncertainties have been widely studied, namely aleatoric uncertainty and epistemic uncertainty [22]. Aleatoric uncertainty captures the natural randomness in data arising from noise in the data collection, *e.g.* sensor noise. Epistemic uncertainty captures the lack of representativeness of the model which can be explained away with increasing training data [22].

Aleatoric uncertainty modeling: The basis assumption for aleatoric uncertainty estimation is that the model parameter θ is fixed and unknown, which leads to non-Bayesian Neural Networks based framework. [24] presents a network which yields a probabilistic distribution as output in order to capture such uncertainty. [44] employs a teacher-student paradigm to distill the aleatoric uncertainty. The teacher network generates multiple predicative samples by incorporating aleatoric uncertainty for the student network to learn. [25] uses an adversarial perturbation technique to generate additional training data for the model to capture the aleatoric uncertainty.

Epistemic uncertainty modeling: The epistemic uncertainty estimation models aims to estimate the distribution of the model parameter set $p(\theta|D)$, where the θ follows some specific distribution leading to Bayesian Neural Network (BNN), and D is the training dataset. The main focus of BNN is to achieve effective posterior inference $p(\theta|D)$, which is intractable in practice. In this way, existing techniques mainly work on approximate posterior inference. Among them, Markov Chain Monte Carlo (MCMC) [37] methods have been proposed as an approximation solution. A few of its variants, such as stochastic MCMC [48, 34, 17, 5] are designed to improve its scalability to larger datasets. An alternative approximation solution is through variational inference [49, 32, 3]. Another line of work adopts a sampling based approach [1, 22]. The dropout method [16] derives the confidence from the multiple forward passes of samples. Ensemble based solutions pass the input data through multiple replicated models [25] or a model with a multi-head decoder [53] to obtain multiple results in order to compute the inference mean and variance.

Camouflaged Object Detection: Camouflaged object detection models [28, 13, 26, 58, 52, 42, 9, 27, 12] are designed to discover the entire scope of camouflaged object(s). Different from regular objects that usually have different levels of contrast with their surroundings, camouflaged objects show similar appearance to the environment. [7] observes that an effective camouflage includes two mechanisms: 1) background pattern matching, where the colour is similar to the environment, and 2) disruptive coloration, which usually involves bright colours along edge, obscuring the boundary between camouflaged object and the background. To detect camouflaged objects, [28] introduces a multi-task learning network with a segmenta-

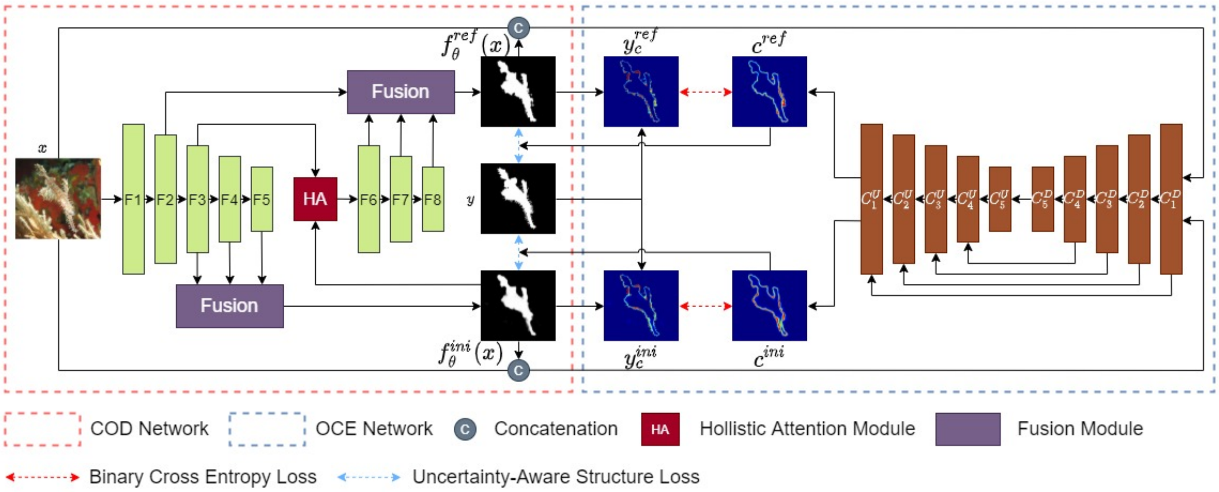

Figure 2. The proposed confidence-aware camouflaged object detection network (CANet) is composed of two interdependent networks. The dynamic confidence supervision is derived from the predicted result of the COD network and the ground-truth camouflage map. The output of the confidence estimation network is used to guide the COD network to focus on learning image parts with low confidences through the uncertainty-guided structure loss. $F_i(i = 1, ..., 8)$ denotes the feature maps of the camouflaged object detection network. $C_i^D(i = 1, ..., 5)$ and $C_i^U(i = 1, ..., 5)$ denote the feature maps associated with the down convolution and up convolution operations of the confidence estimation network separately. y and $f_\theta(x)$ denote the ground-truth and predicted camouflage maps. y_C and c denote the dynamic confidence supervision and the predicted confidence maps.

tion module to produce the camouflage map and a classification module to estimate the possibility of the input image containing camouflaged objects. [13] contributes the largest camouflaged object training set with an SINet for camouflaged object detection. [33] designs a triple task learning framework to simultaneously detect, localize and rank the camouflaged objects. Different from existing techniques, we introduce confidence-aware camouflaged object detection by modelling the pixel-wise difficulty of camouflaged object detection with extra uncertainty maps produced.

Uncertainty-Aware Learning: With the estimated uncertainty map, one can use it to achieve difficulty-aware learning, which has shown to be effective in improving model performance. [38] utilises the estimated confidence to relax the softmax loss function to achieve better performance in pedestrian detection. [39] uses the learnt confidence to pick out hard pixels and directs the segmentation model to focus on them. [46] employs the estimated confidence as an additional filter on the pixel-adaptive convolution to improve the performance of the upsampling operation. Focal loss [30] emphasises learning hard samples in the classification task to deal with the imbalanced learning problem.

Uniqueness of our solution: Our OCENet differs from the existing aleatoric uncertainty modeling methods by directly learning from the difference between prediction and ground truth. Conventional strategies for aleatoric uncertainty modeling involve [22, 23] no supervision for the aleatoric uncertainty, which is only introduced as weight and regularizer to the task related loss function. We dynamically generate uncertainty map to direct the CODNet to put more emphasis on learning areas where predictions are regarded as uncertain. These weights are learnt specifically for each sample rather than assigned universally to the entire dataset.

3. Our Method

3.1. Overview

As a binary segmentation network, camouflaged object detection models usually follow the conventional practice of regressing the camouflage map given the input image [28, 13]. We introduce a mutual-supervising camouflaged object detection learning framework to directly model the aleatoric uncertainty. Two main modules are included in our framework, namely a Camouflaged Object Detection Network (CODNet) to produce the camouflage map, and an Online Confidence Estimation Network (OCENet) to explicitly estimate the aleatoric uncertainty in the current prediction. We show the pipeline of our framework in Fig. 2.

Our training dataset is $D = \{x_n, y_n\}_{n=1}^N$, where x_n and y_n are the image and its corresponding ground-truth camouflage map, n indexes the training images, and N is size of the training dataset. We define the CODNet as f_θ which generates our predicted camouflage map. Then OCENet, g_β, takes the concatenation of the predicted camouflage map and image as input to estimate the pixel-wise uncertainty map indicating the awareness of the model towards the prediction of CODNet.

3.2. Camouflaged Object Detection Network

The proposed CODNet employs a ResNet-50 [19] encoder to produce the feature maps $F_i (i = 1, ..., 5)$. A Fusion Module (FM) is proposed to combine the feature maps of different levels. As illustrated in Fig. 3, the FM progressively fuses high-level features with the lower-level features. In each fusion operation, the highest-level feature is included to provide semantic guidance. Similar to [50], the initial prediction $\hat{y}^{ini} = f_\theta^{ini}(x)$ utilising feature maps F_{3-5} also serves as an attention mechanism on feature map F_3, leading to the computation of feature map F_6. $F_{7,8}$ are obtained by passing F_6 through residual blocks. The final prediction $\hat{y}^{ref} = f_\theta^{ref}(x)$ is computed by fusing the feature maps $F_{2,6-8}$. The relatively low-level feature map F_2 provides more spatial information which is important for segmentation tasks to recover a more crisp structure.

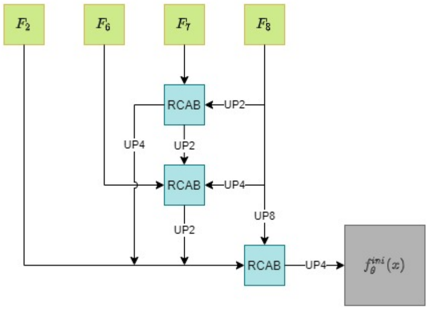

Figure 3. The structure of the fusion module used to produce the final predicted camouflage map. RCAB is the residual channel attention block from [55]. UP denotes the upsampling through bilinear interpolation and its suffix indicates the scale factor.

Given an input image x, our camouflaged object detection network produces two different predictions: $\hat{y}^{ini}$ and $\hat{y}^{ref}$ in the range of $(0, 1)$, where supervision is provided for both predictions. This setting allows the initial prediction to recover a more complete camouflaged object, which subsequently serves as a better attention map to filter the feature map F_3. The final prediction $\hat{y}^{ref}$ is adopted as the camouflaged object detection result for evaluation.

3.3. Online Confidence Estimation Network

OCENet employs a U-Net [43] structure to obtain pixel-accurate uncertainty prediction. It consists of 5 down-convolution features denoted as $C_i^D (i = 1, ..., 5)$ and 5 up-convolution features denoted as $C_i^U (i = 1, ..., 5)$ with pairwise corresponding resolutions. The proposed down-convolution block has two 3×3 convolutional layers ("Conv3"), each followed by a batch normalisation [21] and a Leaky ReLU [18] activation function with negative slope set to 0.2 and ends with a dropout layer ("$D(\cdot)$") of rate 0.5. The down-convolution operation can be summarised as in Eq. 1:

$$C_n^D = D(Conv3(Conv3(C_{n-1}^D))) \tag{1}$$

The up-convolution block consists of a 2×2 transposed convolutional layer ("TConv2") and two 3×3 convolutional layers, each followed by a batch normalisation and a Leaky ReLU activation function with 0.2 negative slope. Down-convolution and up-convolution features are concatenated before the two convolutional layers. Dropout layers of rate 0.5 are used after the transposed convolutional layer and at the end of the up-convolution operation. The up-convolution operation can be summarised as in Eq. 2:

$$C_n^U = D(Conv3(Conv3(\amalg(C_n^D, D(TConv2(C_{n+1}^U)))))) \tag{2}$$

where $\amalg(\cdot)$ denotes a concatenation operation.

CODNet takes the concatenation of model prediction ($\hat{y}^{ini}$ and $\hat{y}^{ref}$) and image x as input to produce an one-channel confidence map, which is defined as $c^{ini} = g_\beta(\amalg(x, \hat{y}^{ini}))$ for the initial prediction, and $c^{ref} = g_\beta(\amalg(x, \hat{y}^{ref}))$ for the final prediction. The estimated confidence maps are supervised with dynamic uncertainty supervision derived from the predictions of the camouflaged object detection network $f_\theta(x)$ and the ground-truth camouflage map y.

3.4. Dynamic Uncertainty Supervision

Existing methods [22, 40] model aleatoric uncertainty as variance $\sigma(x)^2$ as shown in Eq. 3[1].

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{n=1}^{N}(\frac{1}{2\sigma(x_n)^2}\|p_i - y_i\|_2 + \frac{1}{2}log(\sigma(x_n)^2)), \tag{3}$$

where N is size of the training dataset, x_n is the input image with n indexes the images, θ is model parameter set, and p_i and y_i are i^{th} prediction and groundtruth respectively. The unbounded variance is employed to balance the loss. It is maximised to reduce the loss incurred by the L2 loss on the wrong predictions, and regularised to prevent it from becoming infinitely large. Instead, we use the difference between the prediction and groundtruth as explicit supervision to model the aleatoric uncertainty. In our work, it represents uncertainty in the prediction conditioned on the input image.

We derive the dynamic uncertainty supervision y_c for the OCENet defined as in Eq. 4:

$$y_c = y \times (1 - \hat{y}) + (1 - y) \times \hat{y}. \tag{4}$$

The dynamic uncertainty supervision y_c is defined as a pixel-wise L1 distance between the prediction $f_\theta(x)$ and its corresponding ground-truth label y. It has high uncertainty assigned to pixels where the camouflaged object

[1] This aleatoric uncertainty-aware loss function is based on Gaussian likelihood.

detection network makes confident but false predictions. For example, if the camouflage prediction for pixel u, v is $\hat{y}^{u,v} = 0.01$, indicating a background pixel, whereas its ground-truth label is $y^{u,v} = 1$, suggesting it is a foreground pixel, our dynamic supervision is $y_c^{u,v} = 0.99$ representing an uncertain or difficult pixel.

OCENet is trained with a binary cross-entropy loss which is defined as in Eq. 5:

$$\mathcal{L}_c = 0.5 \times (\mathcal{L}_{ce}(c^{ini}, y_c^{ini}) + \mathcal{L}_{ce}(c^{ref}, y_c^{ref})), \qquad (5)$$

where $\mathcal{L}_{ce}$ is the binary cross-entropy loss, y_c^{ini} and y_c^{ref} are dynamic supervisions for the initial prediction and our final prediction respectively.

Algorithm 1 Confidence-aware Camouflaged Object Detection

```
Input:
(1) Training dataset $D = \{x_n, y_n\}_{n=1}^N$, where $N$ is size of the training dataset;
(2) maximal number of learning epochs $E$.
Output: Parameters $\theta$ for the camouflaged object detection module (CODM) and parameters $\beta$ for the confidence estimation module (CEM).
1: Initialise $\theta$ and $\beta$
2: for $t \leftarrow 1$ to $E$ do
3:     Generate camouflage predictions $\hat{y}^{ini} = f_\theta^{ini}(x)$ and $\hat{y}^{ref} = f_\theta^{ref}(x)$ from the CODM.
4:     Produce dynamic supervisions $y_c^{ini}$ and $y_c^{ref}$ for the CEM with Eq. 4.
5:     Obtain the confidence maps $c^{ini} = g_\beta(\amalg(\hat{y}^{ini}, x))$ and $c^{ref} = g_\beta(\amalg(\hat{y}^{ref}, x))$ from the CEM.
6:     Update CEM with loss function in Eq. 5.
7:     Generate confidence-aware weight $\omega^{ini} = 1 + \lambda c^{ini}$ for $\hat{y}^{ini}$ and confidence-aware weight $\omega^{ref} = 1+\lambda c^{ref}$ for $\hat{y}^{ref}$.
8:     Update CODM with loss function in Eq. 6.
```

3.5. Uncertainty-Aware Learning

Camouflaged object detection has different learning difficulties across the image. The pixels along the object boundary are harder to differentiate than the background pixels that are further away from the camouflaged objects. Further, the camouflage foreground contains parts with different level of camouflage, where some parts are easy to recognise, *e.g.* eyes, mouths and *etc*. and some others are hard to distinguish, *e.g.* the body region has similar appearance to the background. We intend to model such varying learning difficulty across the image by modeling the uncertainty awareness in our CODNet. Specifically, inspired by [47], we propose to train the camouflaged object detection network with an uncertainty-aware structure loss, which is defined in Eq. 6:

$$\mathcal{L}_s = \sum_{u,v} w^{u,v} \mathcal{L}_{ce} + \sum_{u,v} w^{u,v} \mathcal{L}_{dice}, \qquad (6)$$

where the weight term is defined as: $w^{ini} = 1 + \lambda c^{ini}$ for the initial prediction $f_\theta^{ini}(x)$ and $w^{ref} = 1 + \lambda c^{ref}$ for our final prediction $f_\theta^{ref}(x)$, and λ is a parameter controlling the scale of attention given to uncertain pixels. We emperically set $\lambda = 10$ to achieve the best performance. The first term is a weighted binary cross-entropy loss and the second term is a weighted Dice Loss. The weight term w provides sample specific pixel-wise weights, letting the CODNet focus on learning uncertain pixels, especially where confident false predictions are made. Our whole algorithm is shown in Algorithm 1. The comparisons between predictions with and without confidence as guidance in Fig. 1 show the effectiveness of our confidence-aware learning.

4. Experimental Results

4.1. Setting:

Dataset: We train our model using the COD10K training set [13], and test on four camouflaged object detection testing sets, including the CAMO [28], CHAMELEON [45], COD10K testing dataset [13] and NC4K dataset [33].

Evaluation Metrics: We use four evaluation metrics to evaluate the performance of the camouflaged object detection models, including Mean Absolute Error ($\mathcal{M}$), Mean F-measure (F_β), Mean E-measure [11] (E_ξ) and S-measure [10] (S_α). A detailed introduction to those metrics appears in the supplementary materials.

Training details: We train our model in Pytorch with ResNet-50 [19] as backbone, where the encoder part is initialized with weights trained on ImageNet, and other newly added layers are randomly initialized. We resize all the images and ground truth to 480×480. The maximum epoch is 50. The initial learning rates are 2.5×10^{-5} and 1.5×10^{-5} for the camouflaged object detection network and confidence estimation network respectively. The whole training takes 8.5 hours with batch size 10 on two NVIDIA GTX 2080Ti GPUs.

4.2. Performance comparison

As there are few COD models, we retrain state-of-the-art salient object detection (SOD) methods [50, 51, 31, 47, 57, 41, 56, 14, 4] on the COD10K training dataset [13] to achieve part of the COD benchmark models in Table 1.

Quantitative comparison: We show performance of the compared methods Tab. 1. It can be seen that our proposed confidence-aware camouflaged object detection network compares favourably against the previous state-of-the-art methods on all four datasets. The improvements over SINet [13] are most significant on Mean Absolute Error evaluation, which ranges between $14.7 - 18.5\%$ on the four datasets. Among the salient object detection methods, RASNet [4] obtains comparable performance with SINet [13] although it is not designed specifically for the cam-

Table 1. Performance comparison with state-of-the-art methods,

Method	Year	BkB	CAMO [28] $S_\alpha \uparrow$	CAMO [28] $F_\beta \uparrow$	CAMO [28] $E_\xi \uparrow$	CAMO [28] $\mathcal{M} \downarrow$	CHAMELEON [45] $S_\alpha \uparrow$	CHAMELEON [45] $F_\beta \uparrow$	CHAMELEON [45] $E_\xi \uparrow$	CHAMELEON [45] $\mathcal{M} \downarrow$	COD10K [13] $S_\alpha \uparrow$	COD10K [13] $F_\beta \uparrow$	COD10K [13] $E_\xi \uparrow$	COD10K [13] $\mathcal{M} \downarrow$	NC4K [33] $S_\alpha \uparrow$	NC4K [33] $F_\beta \uparrow$	NC4K [33] $E_\xi \uparrow$	NC4K [33] $\mathcal{M} \downarrow$
CPD [50]	2019	VGG-16	0.716	0.618	0.723	0.113	0.857	0.771	0.874	0.048	0.750	0.595	0.776	0.053	0.790	0.708	0.810	0.071
SCRN [51]	2019	ResNet-50	0.779	0.705	0.796	0.090	0.876	0.787	0.889	0.042	0.789	0.651	0.817	0.047	0.832	0.759	0.855	0.059
PoolNet [31]	2019	ResNet-50	0.730	0.643	0.746	0.105	0.845	0.749	0.864	0.054	0.740	0.576	0.776	0.056	0.785	0.699	0.814	0.073
BASNet [41]	2019	ResNet-34	0.615	0.503	0.671	0.124	0.847	0.795	0.883	0.044	0.661	0.486	0.729	0.071	0.698	0.613	0.761	0.094
EGNet [56]	2019	ResNet-50	0.737	0.655	0.758	0.102	0.856	0.766	0.883	0.049	0.751	0.595	0.793	0.053	0.796	0.718	0.830	0.067
F3Net [47]	2020	ResNet-50	0.711	0.616	0.741	0.109	0.848	0.770	0.894	0.047	0.739	0.593	0.795	0.051	0.782	0.706	0.825	0.069
ITSD[57]	2020	ResNet-50	0.750	0.663	0.779	0.102	0.814	0.705	0.844	0.057	0.767	0.615	0.808	0.051	0.811	0.729	0.845	0.064
SINet [13]	2020	ResNet-50	0.745	0.702	0.804	0.092	0.872	0.827	0.936	0.034	0.776	0.679	0.864	0.043	0.810	0.772	0.873	0.057
R2Net [14]	2020	VGG-16	0.772	0.685	0.777	0.098	0.861	0.766	0.869	0.047	0.787	0.636	0.801	0.048	0.823	0.739	0.835	0.064
RASNet [4]	2020	VGG-16	0.763	0.716	0.824	0.090	0.857	0.804	0.923	0.040	0.778	0.673	0.865	0.044	0.817	0.772	0.880	0.057
MGL [52]	2021	ResNet-50	0.775	0.673	0.847	0.088	0.893	0.813	0.923	0.030	0.814	0.666	0.865	0.035	-	-	-	-
TINet [58]	2021	ResNet-50	0.781	0.678	0.847	0.087	0.874	0.783	0.916	0.038	0.793	0.635	0.848	0.043	-	-	-	-
PFNet [35]	2021	ResNet-50	0.782	0.695	0.852	0.085	0.882	0.810	0.942	0.033	0.800	0.660	0.868	0.040	-	-	-	-
LSR [33]	2021	ResNet-50	0.793	0.725	0.826	0.085	0.893	0.839	0.938	0.033	0.793	0.685	0.868	0.041	0.839	0.779	0.883	0.053
JSCOD [29]	2021	ResNet-50	0.803	0.759	0.853	0.076	0.894	**0.848**	**0.943**	0.030	0.817	0.726	**0.892**	0.035	-	-	-	-
Ours	2021	ResNet50	**0.807**	**0.767**	**0.866**	**0.075**	**0.901**	0.843	0.940	**0.028**	**0.832**	**0.745**	0.890	**0.032**	**0.857**	**0.817**	**0.899**	**0.044**

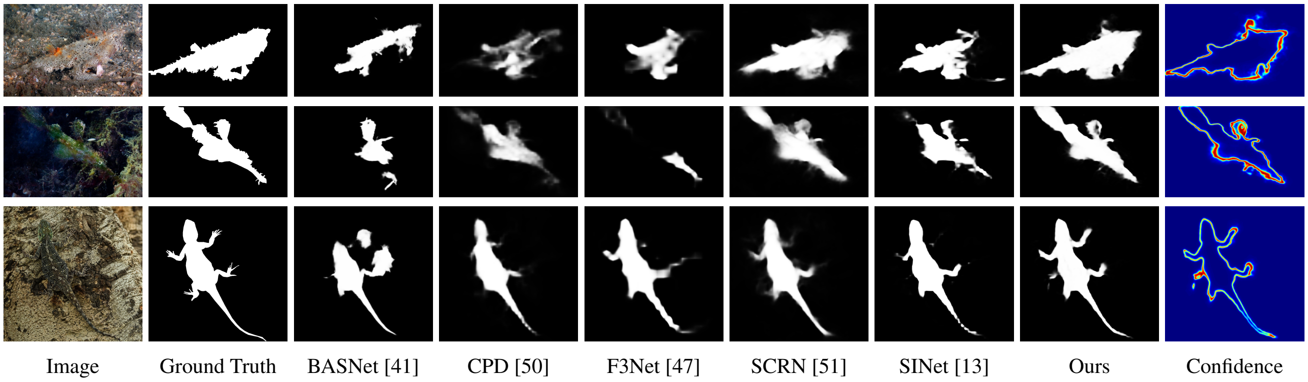

Figure 4. Predictions of our method and those compared methods.

Table 2. Performance comparison of ablation study models.

Method	CAMO [28] $S_\alpha \uparrow$	CAMO [28] $F_\beta \uparrow$	CAMO [28] $E_\xi \uparrow$	CAMO [28] $\mathcal{M} \downarrow$	CHAMELEON [45] $S_\alpha \uparrow$	CHAMELEON [45] $F_\beta \uparrow$	CHAMELEON [45] $E_\xi \uparrow$	CHAMELEON [45] $\mathcal{M} \downarrow$	COD10K [13] $S_\alpha \uparrow$	COD10K [13] $F_\beta \uparrow$	COD10K [13] $E_\xi \uparrow$	COD10K [13] $\mathcal{M} \downarrow$	NC4K [33] $S_\alpha \uparrow$	NC4K [33] $F_\beta \uparrow$	NC4K [33] $E_\xi \uparrow$	NC4K [33] $\mathcal{M} \downarrow$
M1	0.780	0.751	0.858	0.080	0.862	0.794	0.918	0.031	0.791	0.667	0.864	0.037	0.828	0.778	0.893	0.048
M2	0.794	0.767	0.859	0.076	0.881	0.819	0.926	0.031	0.808	0.700	0.881	0.036	0.839	0.802	0.900	0.047
M3	0.798	0.767	0.866	0.080	0.880	0.821	0.933	0.031	0.807	0.694	0.875	0.037	0.840	0.799	**0.900**	0.048
Ours	**0.807**	**0.767**	**0.866**	**0.075**	**0.901**	**0.843**	**0.940**	**0.028**	**0.832**	**0.745**	**0.890**	**0.032**	**0.857**	**0.817**	0.899	**0.044**

ouflaged object detection task. However, it is still outperformed by our proposed method on all evaluation metrics. We notice the relatively similar S-measure and mean F-measure of our solution compared with LSR [33] on the CHAMELEON [45] dataset. This mainly due to the small size of the CHAMELEON [45] dataset, with 76 samples in total. The performance gap on MAE further indicates effectiveness of our solution.

Qualitative comparison: We show predictions of our method and compared methods in Fig. 4. In the first and second rows, [41, 50, 47, 13] fail to recover the main structure of the *Batfish* and *Ghost Pipefish*. [51] can only discover the main body while predictions around the object boundary are ambiguous and incorrect. On the contrary, our method is able to segment more complete camouflaged objects whose boundaries are closer to those of ground truths. On the third row, [41, 50, 47, 13, 51] recover the main body of the *Lizard*, but they fail to find the limbs. In comparison, our method successfully segments both the main body and the four feet, both of which are close to the ground truth. Complementing to our camouflaged object detection, our estimated confidence map picks up inaccurate predictions associated with both over-segmentation and under-segmentation issues at the object boundary.

Inference time comparison: Different from the compared methods, which produce a single camouflage map in the end, we introduce a confidence estimation module to evaluate pixel-wise awareness of model of the predictions. Al-

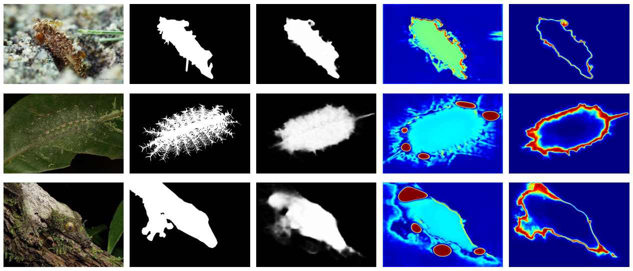

Figure 5. Comparison of confidence maps produced with dynamic supervision and adversarial learning setting. From left to right are image, ground truth map, model prediction, confidence with adversarial learning and confidence with our dynamic supervision.

though the extra module is included in our framework, our inference time is 0.0211s per image for the camouflaged object detection network and 0.0094s per image for the confidence estimation network, which is comparable with existing techniques, such as 0.0296s of SINet [13].

4.3. Ablation study

We have two main modules in our confidence-aware camouflaged object detection network, namely a camouflaged object detection network and a confidence estimation network. We perform the following ablation study to examine the contribution of the main components of our framework. We show performance of these models in Tab. 2.

The structure of the camouflaged object detection network: We adopt the holistic attention module in [50] to refine the module prediction with the initial prediction as attention. To test how our model performs without the holistic attention module, we train the camouflaged object detection network with only the initial prediction as output and denote it as "M1". Further, we add the holistic attention module to "M1" and obtain "M2". Tab. 2 shows that "M2" consistently improves over "M1" on all evaluation metrics, demonstrating that the holistic attention is able to help the model extract more discriminative features.

Joint training of "M2" and the confidence estimation network: We add the confidence estimation network g_β to "M2" without difficulty-aware learning. As there exists no interaction between the confidence estimation network and the camouflaged object detection network, the resulting model achieve same performance as "M2".

The supervision of the confidence estimation network: Similar to [20, 39], another option to generate supervision for the confidence estimation module is to assign 0 for the prediction and 1 for the ground truth map following the adversarial learning pipeline. We perform this experiment[2] and show its results as "M3" in Table 2. In this setting, we regard our confidence estimation network as a discriminator and our camouflaged object detection network as a generator. A well-trained discriminator should converge to 0.5 indicating it cannot distinguish the prediction from the ground truth. Therefore, we define the estimated confidence in this adversarial learning setting as:

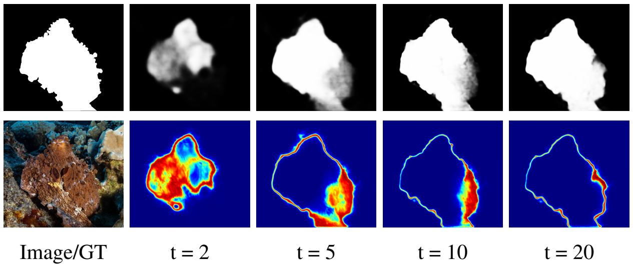

Figure 6. Using the estimated confidence map as an indicator of the prediction quality. The first column displays the ground truth and image. Predictions and corresponding uncertainty maps from different training stages are displayed from the second to the fifth column. Red indicates low confidence and blue indicates high confidence. t indicates the training epoch.

$$\hat{y}_c = \frac{|g_\beta(\Pi(x, f_\theta(x))) - 0.5|}{0.5}. \tag{7}$$

A trivial solution exists with above discriminator supervision, that is the model will simply project hard samples ($y \in \{0,1\}$ to 1) to 1 and soft samples ($f_\theta(x) \in (0,1)$) to 0. To prevent this, we introduce a label perturbation technique which relaxes the ground-truth labels from $\{0,1\}$ to $\{v \mid 0 < v < 0.01$ or $0.99 < v < 1\}$ corresponding to the background label and the foreground label respectively. Confident correct predictions on pixels of the camouflaged object detection network are associated with values between these ranges, resulting in estimating high confidences in these pixels, while pixels with moderate scores, *e.g.* 0.4 for weak background prediction or 0.6 for foreground prediction, are assigned high uncertainties.

Experimental results in Tab. 2 show that "Ours" in general outperforms "M3", indicating that the confidence estimation network trained with dynamic supervision produces more reliable confidence maps, leading to better performance with the difficulty-aware learning in Eq.6.

4.4. Discussion

Comparison with adversarial learning: Fig. 5 illustrates the difference between the confidence maps produced with the adversarial learning setting and dynamic supervision method. In general, the confidence map produced with the adversarial learning setting is biased to have higher uncertainty values associated with the foreground predictions. These biased uncertainties are consistent across the foreground predictions although they are mostly correct. For

[2] Please refer to the supplementary material for the network overview and implementation details.

Table 3. Performance comparison of choosing different λ values.

	CAMO [28]				CHAMELEON [45]				COD10K [13]				NC4K [33]			
Method	$S_\alpha \uparrow$	$F_\beta \uparrow$	$E_\xi \uparrow$	$\mathcal{M} \downarrow$	$S_\alpha \uparrow$	$F_\beta \uparrow$	$E_\xi \uparrow$	$\mathcal{M} \downarrow$	$S_\alpha \uparrow$	$F_\beta \uparrow$	$E_\xi \uparrow$	$\mathcal{M} \downarrow$	$S_\alpha \uparrow$	$F_\beta \uparrow$	$E_\xi \uparrow$	$\mathcal{M} \downarrow$
$\lambda = 3$	0.802	0.755	0.841	0.080	0.890	0.828	0.929	0.032	0.823	0.728	0.883	0.036	0.850	0.804	0.892	0.049
$\lambda = 5$	0.805	0.768	0.851	0.077	0.901	0.840	0.933	0.029	0.834	0.743	0.887	0.033	0.857	0.813	0.896	0.046
λ_D	0.804	0.764	0.862	**0.074**	0.875	0.810	0.923	0.034	0.807	0.697	0.877	0.037	0.843	0.798	**0.900**	0.047
Ours ($\lambda = 10$)	**0.807**	**0.767**	**0.875**	0.075	**0.901**	**0.843**	**0.940**	**0.028**	**0.832**	**0.745**	**0.890**	**0.032**	**0.857**	**0.817**	0.899	**0.044**

example, in the samples presented in Fig. 5, the strong foreground predictions inside the object are correct, thus should be regarded as confident. The desired uncertainty values are manifested in the confidence map produced with dynamic supervision. Correct foreground predictions are assigned with high confidence. The errors only occur at the boundary pixels, which correspond to high uncertainty values.

Confidence maps produced with the adversarial learning setting generate artifacts, a blob of uncertain area centred at weak foreground predictions. Although these artifacts locate the uncertain foreground predictions, they fail to provide spatially-accurate uncertainties. On the contrary, the confidence map with dynamic supervision is able to delineate a more precise uncertainty structure. On the second row of Fig. 5, it produces a thickened uncertainty prediction along the predicted object boundaries where most errors occur. Just inside these boundaries, it faintly traces the thin body parts of the caterpillar at the top-right corner and the left side. The structure-preserving property of the confidence map with dynamic supervision is best demonstrated on the third sample of Fig. 5. Although the camouflaged object detection network fails to predict the webbed frog foot, its structure is picked up by the confidence map produced with dynamic supervision where the boundary of the webbed foot is delineated, forcing the camouflaged object detection network to focus on learning the missing parts.

When the uncertainty map is used as guidance to the structure loss, the adversarial learning version is able to direct attention to weak prediction areas where errors are prone to occur. However, despite its localisation capability, it is not pixel-wise accurate. On the contrary, the dynamic supervision version can discover object structure that the camouflaged object detection network fails to find. It complements the camouflaged object detection network, refining object structure and recovering initially lost object parts.

Confidence module as a trained evaluation tool without relying on ground truth maps: Our confidence map can serve as a rough evaluation tool of the prediction quality of the camouflaged object detection network without relying on the ground-truth segmentation map. Fig. 6 illustrates the predicted camouflage map and its corresponding confidence map of a sample at different stages of training. The sample is regarded as hard and its initial prediction discovers only a small part of foreground object at the second epoch. This leads to large areas of high uncertainty values in its corresponding confidence map. As the prediction becomes more refined as the training progresses, the high-uncertainty areas in the confidence maps shrink as a result, eventually highlighting only the structures of the camouflaged objects where errors are prone to occur. In addition, Fig. 6 also validates that our estimated confidence map guides the camouflaged object detection network to gradually recover the initially lost object parts of the hard samples.

Hyper-parameter analysis: The impact of selecting different values of λ for $\omega^{u,v}$ in Eq. 6, which is a factor controlling the uncertainty guidance in the structure loss, is demonstrated in Tab. 3. We ablate $\lambda = 3, 5, \lambda_D$ where λ_D is a dynamic factor defined as $\lambda_D = \min\{2 \times ReLU(t-5), 20\}$, where t is the current training epoch. The results show that our results of defining $\lambda = 10$ achieve better performance. The inferior performance of the dynamic factor λ_D can be attributed to that it provides insufficient guidance in the early stage of training. As it is difficult to tune it to achieve adaptive uncertainty weighting, in this paper, we define fixed λ for the entire training stage, which is proven to work better in general.

5. Conclusion

We introduce an on-line aleatoric uncertainty estimation technique for camouflaged object detection. The conventional approach to aleatoric uncertainty modeling involves only supervision for the task related loss function as shown in Eq. 3. In this paper, we deal with on-line aleatoric uncertainty estimation and introduce dynamic supervision for the aleatoric uncertainty estimation module to highlight the wrongly predicted areas. Specifically, our framework is composed of an interdependent camouflaged object detection network (CODNet) and an on-line confidence estimation network (OCENet). The dynamic confidence label is generated to train the OCENet, which is derived from the prediction of the CODNet and the ground truth map. The estimated confidence map from the OCENet directs the CODNet to place more emphasis on learning areas with uncertain predictions. Our proposed network performs favourably against existing camouflaged object detection methods on four benchmark camouflaged object detection testing datasets. Further, the generated confidence map provides an effective solution to explain the model prediction without relying on the ground truth map.

Acknowledgments. This research was in-part supported by the ANU-Optus Bushfire Research Center of Excellence.

References

[1] Murat Seckin Ayhan and Philipp Berens. Test-time data augmentation for estimation of heteroscedastic aleatoric uncertainty in deep neural networks. In *International conference on Medical Imaging with Deep Learning*, 2018.

[2] Alexandra Barbosa, Lydia M Mäthger, Kendra C Buresch, Jennifer Kelly, Charles Chubb, Chuan-Chin Chiao, and Roger T Hanlon. Cuttlefish camouflage: the effects of substrate contrast and size in evoking uniform, mottle or disruptive body patterns. *Vis. Res.*, 48(10):1242–1253, 2008.

[3] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *Int. Conf. Mach. Learn.*, pages 1613–1622. PMLR, 2015.

[4] Shuhan Chen, Xiuli Tan, Ben Wang, Huchuan Lu, Xuelong Hu, and Yun Fu. Reverse attention-based residual network for salient object detection. *IEEE T. Image Process.*, 29:3763–3776, 2020.

[5] Tianqi Chen, Emily Fox, and Carlos Guestrin. Stochastic gradient hamiltonian monte carlo. In *Int. Conf. Mach. Learn.*, pages 1683–1691. PMLR, 2014.

[6] Hugh Bamford Cott. Adaptive coloration in animals. *Nature*, 1940.

[7] 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):72–74, 2005.

[8] Yukun Ding, Jinglan Liu, Xiaowei Xu, Meiping Huang, Jian Zhuang, Jinjun Xiong, and Yiyu Shi. Uncertainty-aware training of neural networks for selective medical image segmentation. In *Medical Imaging with Deep Learning*, pages 156–173. PMLR, 2020.

[9] Bo Dong, Mingchen Zhuge, Yongxiong Wang, Hongbo Bi, and Geng Chen. Towards accurate camouflaged object detection with mixture convolution and interactive fusion. *arXiv preprint arXiv:2101.05687*, 2021.

[10] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *Int. Conf. Comput. Vis.*, pages 4548–4557, 2017.

[11] Deng-Ping Fan, Cheng Gong, Yang Cao, Bo Ren, Ming-Ming Cheng, and Ali Borji. Enhanced-alignment measure for binary foreground map evaluation. *arXiv preprint arXiv:1805.10421*, 2018.

[12] Deng-Ping Fan, Ge-Peng Ji, Ming-Ming Cheng, and Ling Shao. Concealed object detection. *IEEE T. Pattern Anal. Mach. Intell.*, 2021.

[13] Deng-Ping Fan, Ge-Peng Ji, Guolei Sun, Ming-Ming Cheng, Jianbing Shen, and Ling Shao. Camouflaged object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2777–2787, 2020.

[14] Mengyang Feng, Huchuan Lu, and Yizhou Yu. Residual learning for salient object detection. *IEEE T. Image Process.*, 29:4696–4708, 2020.

[15] Peter Forbes. *Dazzled and deceived: mimicry and camouflage*. Yale University Press, 2011.

[16] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Int. Conf. Mach. Learn.*, pages 1050–1059. PMLR, 2016.

[17] Wenbo Gong, Yingzhen Li, and José Miguel Hernández-Lobato. Meta-learning for stochastic gradient mcmc. *arXiv preprint arXiv:1806.04522*, 2018.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Int. Conf. Comput. Vis.*, pages 1026–1034, 2015.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 770–778, 2016.

[20] W.-C. Hung, Y.-H. Tsai, Y.-T. Liou, Y.-Y. Lin, and M.-H. Yang. Adversarial learning for semi-supervised semantic segmentation. In *Brit. Mach. Vis. Conf.*, 2018.

[21] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Int. Conf. Mach. Learn.*, pages 448–456. PMLR, 2015.

[22] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *arXiv preprint arXiv:1703.04977*, 2017.

[23] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7482–7491, 2018.

[24] Lingkai Kong, Jimeng Sun, and Chao Zhang. Sde-net: Equipping deep neural networks with uncertainty estimates. *arXiv preprint arXiv:2008.10546*, 2020.

[25] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *arXiv preprint arXiv:1612.01474*, 2016.

[26] Hala Lamdouar, Charig Yang, Weidi Xie, and Andrew Zisserman. Betrayed by motion: Camouflaged object discovery via motion segmentation. In *Asi. Conf. Comput. Vis.*, 2020.

[27] Trung-Nghia Le, Yubo Cao, Tan-Cong Nguyen, Minh-Quan Le, Khanh-Duy Nguyen, Thanh-Toan Do, Minh-Triet Tran, and Tam V Nguyen. Camouflaged instance segmentation in-the-wild: Dataset and benchmark suite. *arXiv preprint arXiv:2103.17123*, 2021.

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

[29] Aixuan Li, Jing Zhang, Yunqiu Lv, Bowen Liu, Tong Zhang, and Yuchao Dai. Uncertainty-aware joint salient object and camouflaged object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 10071–10081, 2021.

[30] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Int. Conf. Comput. Vis.*, pages 2980–2988, 2017.

[31] Jiang-Jiang Liu, Qibin Hou, Ming-Ming Cheng, Jiashi Feng, and Jianmin Jiang. A simple pooling-based design for real-time salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3917–3926, 2019.

[32] Christos Louizos and Max Welling. Multiplicative normalizing flows for variational bayesian neural networks. In *Int. Conf. Mach. Learn.*, pages 2218–2227. PMLR, 2017.

[33] Yunqiu Lv, Jing Zhang, Yuchao Dai, Aixuan Li, Bowen Liu, Nick Barnes, and Deng-Ping Fan. Simultaneously localize, segment and rank the camouflaged objects. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021.

[34] Yi-An Ma, Tianqi Chen, and Emily B Fox. A complete recipe for stochastic gradient mcmc. *arXiv preprint arXiv:1506.04696*, 2015.

[35] Haiyang Mei, Ge-Peng Ji, Ziqi Wei, Xin Yang, Xiaopeng Wei, and Deng-Ping Fan. Camouflaged object segmentation with distraction mining. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8772–8781, 2021.

[36] Jooyoung Moon, Jihyo Kim, Younghak Shin, and Sangheum Hwang. Confidence-aware learning for deep neural networks. In *Int. Conf. Mach. Learn.*, pages 7034–7044. PMLR, 2020.

[37] Radford M Neal. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media, 2012.

[38] Lukas Neumann, Andrew Zisserman, and Andrea Vedaldi. Relaxed softmax: Efficient confidence auto-calibration for safe pedestrian detection. In *Adv. Neural Inform. Process. Syst. Worksh.*, 2018.

[39] Dong Nie, Li Wang, Lei Xiang, Sihang Zhou, Ehsan Adeli, and Dinggang Shen. Difficulty-aware attention network with confidence learning for medical image segmentation. In *AAAI Conf. Art. Intell.*, pages 1085–1092, 2019.

[40] David A Nix and Andreas S Weigend. Estimating the mean and variance of the target probability distribution. In *Proceedings of 1994 IEEE International Conference on Neural Networks*, volume 1, pages 55–60, 1994.

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

[42] Jingjing Ren, Xiaowei Hu, Lei Zhu, Xuemiao Xu, Yangyang Xu, Weiming Wang, Zijun Deng, and Pheng-Ann Heng. Deep texture-aware features for camouflaged object detection. *arXiv preprint arXiv:2102.02996*, 2021.

[43] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-assisted Intervention*, pages 234–241. Springer, 2015.

[44] Yichen Shen, Zhilu Zhang, Mert R Sabuncu, and Lin Sun. Real-time uncertainty estimation in computer vision via uncertainty-aware distribution distillation. In *IEEE Winter Conf. Applications of Comput. Vis.*, pages 707–716, 2021.

[45] Przemysław Skurowski, Hassan Abdulameer, Jakub Baszczyk, Tomasz Depta, Adam Kornacki, and Przemysław Kozie. Animal camouflage analysis: Chameleon database. In *Unpublished Manuscript*, 2018.

[46] Anne S Wannenwetsch and Stefan Roth. Probabilistic pixel-adaptive refinement networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 11642–11651, 2020.

[47] Jun Wei, Shuhui Wang, and Qingming Huang. F^{3}net: Fusion, feedback and focus for salient object detection. In *AAAI Conf. Art. Intell.*, pages 12321–12328, 2020.

[48] Max Welling and Yee W Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Int. Conf. Mach. Learn.*, pages 681–688, 2011.

[49] Anqi Wu, Sebastian Nowozin, Edward Meeds, Richard E Turner, Jose Miguel Hernandez-Lobato, and Alexander L Gaunt. Deterministic variational inference for robust bayesian neural networks. *arXiv preprint arXiv:1810.03958*, 2018.

[50] Zhe Wu, Li Su, and Qingming Huang. Cascaded partial decoder for fast and accurate salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3907–3916, 2019.

[51] Zhe Wu, Li Su, and Qingming Huang. Stacked cross refinement network for edge-aware salient object detection. In *Int. Conf. Comput. Vis.*, pages 7264–7273, 2019.

[52] Qiang Zhai, Xin Li, Fan Yang, Chenglizhao Chen, Hong Cheng, and Deng-Ping Fan. Mutual graph learning for camouflaged object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12997–13007, 2021.

[53] Jing Zhang, Yuchao Dai, Xin Yu, Mehrtash Harandi, Nick Barnes, and Richard Hartley. Uncertainty-aware deep calibrated salient object detection. *arXiv preprint arXiv:2012.06020*, 2020.

[54] 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 Conf. Comput. Vis. Pattern Recog.*, pages 8582–8591, 2020.

[55] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Eur. Conf. Comput. Vis.*, pages 286–301, 2018.

[56] 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 *Int. Conf. Comput. Vis.*, pages 8779–8788, 2019.

[57] Huajun Zhou, Xiaohua Xie, Jian-Huang Lai, Zixuan Chen, and Lingxiao Yang. Interactive two-stream decoder for accurate and fast saliency detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9141–9150, 2020.

[58] Jinchao Zhu, Xiaoyu Zhang, Shuo Zhang, and Junnan Liu. Inferring camouflaged objects by texture-aware interactive guidance network. In *AAAI Conf. Art. Intell.*, pages 3599–3607, 2021.