

# Class-Agnostic Segmentation Loss and Its Application to Salient Object Detection and Segmentation

Angira Sharma  
University of Oxford

angira.sharma@cs.ox.ac.uk

Naeemullah Khan  
University of Oxford

naeemullah.khan@eng.ox.ac.uk

Muhammad Mubashar  
LUMS

21100158@lums.edu.pk

Ganesh Sundaramoorthi  
KAUST

ganesh.sundaramoorthi@kaust.edu.sa

Philip Torr  
University of Oxford

philip.torr@eng.ox.ac.uk

## Abstract

In this paper we present a novel loss function, called *class-agnostic segmentation (CAS) loss*. With CAS loss the class descriptors are learned during training of the network. We don't require to define the label of a class a-priori, rather the CAS loss clusters regions with similar appearance together in a weakly-supervised manner. Furthermore, we show that the CAS loss function is sparse, bounded, and robust to class-imbalance. We first apply our CAS loss function with fully-convolutional ResNet101 and DeepLab-v3 architectures to the binary segmentation problem of salient object detection. We investigate the performance against the state-of-the-art methods in two settings of low and high-fidelity training data on **seven** salient object detection datasets. For low-fidelity training data (incorrect class label) class-agnostic segmentation loss outperforms the state-of-the-art methods on salient object detection datasets by staggering margins of around **50%**. For high-fidelity training data (correct class labels) class-agnostic segmentation models perform as good as the state-of-the-art approaches while beating the state-of-the-art methods on most datasets. In order to show the utility of the loss function across different domains we then also test on general segmentation dataset, where class-agnostic segmentation loss outperforms competing losses by huge margins.

## 1. Introduction

Deep learning based methods have achieved state-of-the-art results in many vision applications. The success of these methods is primarily reliant on the fidelity of the datasets they are trained on. For applications like segmentation, recognition and detection, most deep learning based methods use a classification based training approach,

Data	Training Data				Testing			
	Image	Ground Truth	Image	Ground Truth	Image	Ground Truth	CE	CAS
HFD								
LFD								

**Figure 1. Motivation Example:** Traditional segmentation/detection methods work well in HFD setting (high-fidelity training data, where class labels are available and accurate) but fail completely in LFD setting (low-fidelity training data, where class labels are either not available or incorrect). Our Class-Agnostic Segmentation (CAS) loss performs well in both cases, since it is independent of the class label and learns to perform unsupervised clustering of the learned descriptors during training to obtain class labels. Whereas, conventional cross-entropy (CE) based methods completely fail in LFD setting since they are reliant on class label.

where cross-entropy (CE) or a variant of cross-entropy is used to fit the descriptors (network output) to an arbitrarily pre-assigned class label. Generating these class labels on a dataset of sufficiently large size is labour-intensive, restricts research and is prone to human errors. For networks trained with standard loss functions like cross entropy, these labelling errors result in significant degradation of performance (Table 1). We present a class-agnostic segmentation (CAS) loss using which we can train networks independent of these class labels and avoid the consequent degradation of performance.

In segmentation we divide an image into regions of unique statistics. Segmentation has applications in compression, tracking, and recognition. Classical approaches of segmentation either relied on region-based descriptors,

where descriptors were grouped to achieve unique segments, or on edge-based approach, where edges are extracted from an image and post-processing steps like watershed methods are used to obtain region segments.

The current approaches for segmentation and detection, which conventionally use cross-entropy loss or a variant of cross entropy loss, have the following drawbacks: 1) The arbitrarily pre-assigned class label might not necessarily correspond to the representation of the class in the descriptor space. 2) This pre-assignment introduces a hard constraint in labelling the data where the same object has to be labelled with the same label across the entire dataset (thousands of images). For large datasets, data annotation will be performed by a huge pool of moderately trained annotators and there will inevitably be errors in label of objects (since these labels are arbitrary and do not correspond to any fundamental notion of appearance). 3) The number of classes that one can sample is limited and consequently can not be generalised to the infinite number of classes that exist in real life scenes. 4) Using the conventional loss functions the output components of a neural network will have to match the number of classes for these datasets, which will result in very large output vector for large number of classes. 5) The loss functions used are agnostic to the notion of class appearance and simply learn to group similarly labelled objects together.

In this paper we tackle the above challenges. We present a class-agnostic segmentation loss, motivated by metric learning literature [6], [7], which does not need the class label for segmentation, rather the loss's construct is based on unsupervised clustering of learned descriptor to obtain unique segments and relies only on forms of inexpensive ground truth annotations. This allows us to cast the general segmentation problem with deep networks, since we do not need the class labels anymore (general segmentation as opposed to semantic segmentation divides image into unique regions and is not limited to a few classes). Rather the network is constructed to cluster similar appearances together while maximising the inter-cluster distance. This eases the task of annotation (and data generation) as one can use off-the-shelf (edge-based) segmentation methods to semi-automate the data-segmentation task (and assign no class labels whatsoever).

The contributions of our work are: 1) A new class-label agnostic segmentation loss function, which relies on ground truth annotation only and clusters similar segments together by grouping pixels with similar appearance (learned descriptor) together. 2) The class-agnostic segmentation loss function is applied to salient object segmentation and achieves state-of-the-art results despite the fact that we don't use any pre-training or data augmentation that other state-of-the-art methods use. 3) The loss function is applied to general segmentation task and outperforms cross-entropy

and metric learning based losses by a significant margin.

## 2. Related Work

Segmentation methods generally follow one of the two approaches: general approach, where all segments are labelled region-wise based on appearance, and semantic approach, where objects are labelled with class-labels in the dataset. Hence, semantic approach simplifies a general segmentation task to dense classification problem limited to a few classes. Successful segmentation models [3] are based on semantic approach, where the models' abilities are limited by the number of pre-defined classes.

General segmentation algorithms can be broadly classified into region-based methods and edge-based methods. In region-based methods segments are obtained by grouping descriptors together. Traditional region-based methods suffer from inaccurate segmentation results near the boundaries of objects because statistics are aggregated across the boundaries. In [13] better shape-tailored descriptors were introduced to tackle this problem. Building on [13], in [14] continuum scale-space of heat equation was used to obtain coarse-to-fine segmentation, but the descriptors here are still hand-crafted and lack the capacity to discriminate between wide range of textures in natural images.

In edge-based segmentation methods pixels in images are classified to either belong to the edge class or otherwise. Post-processing methods like watershed methods are then applied to the edge maps to obtain region segments. The regions obtained through these approaches are not based on descriptor consistency and hence these methods fails particularly in cases when textures with large textons are present in images [13].

Deep learning methods for general segmentation are primarily edge-based methods. These approaches have been shown to achieve better results [38], [2], [17] on edge detection. Such approaches have used deep networks to derive the probability of a pixel belonging to boundaries between segments. Despite the fact that these approaches achieve impressive results on detecting edges, generating segmentation from edges is still hard and relies on hand-crafted approaches [17], hence, the segmentation problem remains unsolved.

An attempt to cast general segmentation as a region-based learning problem came from [15], where the authors tried to tackle the problem of segmentation by learning a metric to discriminate between shape-tailored descriptors using the Siamese twin networks. However, [15] used fully connected layers to learn the discrimination metric on shape-tailored descriptors computed in pre-processing step. Similar to [15], [18] uses a metric learning scheme to cluster similar pixels together using mean shift algorithm building on [9]. [18] is a metric (and embedding space) learning scheme which can discriminate between pixels pair from

similar region and different regions. These are different contrastive loss based metric learning schemes [6]. Such metric learning approaches are computationally very expensive and require  $\mathcal{O}(N^2)$  terms in the loss where  $N$  is the number of pixels in the image (notice pixel pairs are sampled from images in these methods). Also, these methods suffers severely from class imbalances since the sampling for pixel pair exacerbates the class imbalance. These methods also don't learn a 'class representation' jointly with the metric learning. Our method is novel in terms of learning the 'class representation' jointly with the metric learning without adding any computation overhead. Along similar lines [7] introduced a discriminative loss. Discriminative loss is used for instance segmentation where different instances are clustered in the descriptor space. Discriminative loss does not maximize distance between different classes. Discriminative loss is based on two hyperparameters for inter/intra class variance and the loss is simply a penalty which forces the inter/intra class variance to be close to these hyperparameters. This loss is (can) not (be) used for differentiating different classes, it can only classify different instances and to classify classes [7] uses a cross entropy loss. [26] implements an embedding loss function for instance segmentation, but relies on learning instance specific margin and expensive post-processing step. To the best of our knowledge, ours is the first work to successfully apply a loss based on distance metric learning principles for the task of salient object detection and general segmentation and the first attempt at general region-based segmentation with deep networks where properties of regions are learned rather than a metric for discriminating pixels. Contrary to most metric learning based methods we learn the proxy class label jointly with the descriptor during training and the complexity of our method is  $\mathcal{O}(N)$  where  $N$  is the number of pixels.

Since we apply our class-agnostic loss to the binary segmentation problem of salient object detection, we present a brief literature review of salient object detection here. In supervised salient object detection methods, both input images and ground truth annotations with class labels, are used for training. The class labels are binary, generally, 1 is used to represent salient object and 0 for non-salient objects.

Some state-of-the-art salient detection methods like [11], are based on combining feature maps from different layers of CNN to obtain saliency map. PFAN network [43] uses hand-crafted feature extraction method and channel-wise attention mechanism to extract the most important features in the intermediate layers to generate more accurate saliency maps, however, this results in suboptimal solutions. Other state-of-the-art methods such as PoolNet [22] aggregate high-level information from customised global modules built on top of feature pyramid networks coupled with edge detection at intermediate level of the network. Lat-

est methods like [37] focus on label decoupling framework by focusing on body and edges of salient objects separately; [44] uses multilevel gated units and [45] focuses and utilises correlation between contours and saliency, for salient object detection.

These current state-of-the-art methods require training on large datasets containing real-world data, ideally on cluttered background. However, this kind of data for salient object detection is limited, therefore, majority of the methods such as DeepNet [28] and DeepFix [19] are based on networks pre-trained on the ImageNet [12] dataset coupled with data augmentation [43]. On the contrary, almost all of our models presented in this work do not use any pre-training or data augmentation to keep the complexity minimal.

In semi-supervised and weakly-supervised frameworks there are diverse approaches to solve salient object detection. Some techniques [35] infer potential foreground regions to perform global smooth pooling operation and combine these responses to generate saliency maps. Weak supervision methods, such as CPSNet [42], also rely on multiple sources such as image captions, incomplete or incorrect labels, and multiple images cues, which are passed to more than one networks followed by inter-network feature sharing to output the final saliency map. These type of complex operations result in slow forward computation.

Modified versions of cross entropy loss have also been explored to solve salient object detection. For instance, [27] proposes a variant of cross entropy loss using generative adversarial networks [10]. Another work, BAS-Net [31], proposes a hybrid loss, which is a combination of cross entropy, structural similarity index (SSIM) and intersection over union (IOU). As these loss functions are majorly modified versions of cross entropy loss, they depend on the true class labels for computation. This dependence becomes an obstacle when labelled data is scarce.

**Problem Statement:** We design a loss function which trains the network to clusters pixels of similar appearance together in a weakly-supervised manner. Our loss function forces the descriptors to have low variance on regions/objects, at the same time the descriptor learns to discriminate between different regions.

### 3. Class-Agnostic Segmentation (CAS) Loss

In this section we present the class-agnostic segmentation loss and derive the backpropagation equation when this loss is applied to standard networks.

### 3.1. CAS Loss Function

The class-agnostic segmentation loss is defined as:

$$CAS = \sum_{i=1}^N \int_{r_i} \underbrace{\frac{\alpha \|\mathbf{s}(x) - \hat{\mathbf{s}}(r_i)\|_2^2}{|r_i|}}_{\text{Uniformer}} dx - \sum_{i=1}^N \sum_{\substack{j=1 \\ i \neq j}}^N (1 - \alpha) \underbrace{\|\hat{\mathbf{s}}(r_i) - \hat{\mathbf{s}}(r_j)\|_2^2}_{\text{Discriminator}} \quad (1)$$

where  $N$  is the number of regions in the ground truth mask;  $r_1, \dots, r_i, r_j, \dots, r_N$  denotes the region of the ground truth mask (a particular segment);  $|r_i|$  denotes the number of pixels in the region  $r_i$ ;  $\mathbf{s} = \{s^1, \dots, s^m, \dots, s^M\}$  is a vector of output descriptor components (or softmax output) of the network;  $m \in \{1, \dots, M\}$  where  $M$  denotes the number of output (softmax) channels i.e., number of units in the last layer of the network;  $\alpha \in [0, 1]$  is a scalar, a weighing hyper-parameter which assigns weight to each term; for a region  $r$  we have that,  $\hat{\mathbf{s}}(r) = \{\hat{s}(r)^1, \dots, \hat{s}^m(r), \dots, \hat{s}(r)^M\}$  is a vector containing channel-wise mean of the descriptor values; where for a channel  $m$ ,  $\hat{s}^m(r) = \frac{1}{|r|} \int_r s^m(x) dx$ . In our formulation  $\hat{\mathbf{s}}(r)$  acts as a proxy for class label for region  $r$ .

The uniformer term of the loss function reduces variance of the learned descriptor on the regions (segments). The discriminator term increases distance between the learned descriptors for different regions. These are the two essential properties required of any successful descriptor for segmentation i.e. small intra-class variance and large inter-class discriminability. The uniformer term ensures invariance of a descriptor on a region of interest and the discriminator term ensures that different region have different descriptors. Hence, combination of the two terms trains the model to perform segmentation based on the appearance (rather than class labels). One aspect to note is that we have used squared euclidean distances for both the terms; this is for the simplicity of implementation, but the general framework of class-agnostic segmentation loss will work for any suitable norm.

### 3.2. Gradient of CAS Loss

The gradient of the loss function with respect to the weights  $\omega$  of a deep network is,

$$\nabla_{\omega} CAS = \sum_{i=1}^N \int_{r_i} 2 \frac{\alpha (\mathbf{s}(x) - \hat{\mathbf{s}}(r_i)) (\nabla_{\omega} \mathbf{s}(x) - \nabla_{\omega} \hat{\mathbf{s}}(r_i))}{|r_i|} dx - \sum_{i=1}^N \sum_{\substack{j=1 \\ i \neq j}}^N 2(1 - \alpha) (\hat{\mathbf{s}}(r_i) - \hat{\mathbf{s}}(r_j)) (\nabla_{\omega} \hat{\mathbf{s}}(r_i) - \nabla_{\omega} \hat{\mathbf{s}}(r_j)) \quad (2)$$

From Equation 1 we've,  $\nabla_{\omega} \hat{s}^m(r_i) = \frac{1}{|r_i|} \int_{r_i} \nabla_{\omega} s^m(x) dx$ , where we have  $\nabla_{\omega} s^m(x)$  from the backpropagation of the network as  $s^m(x)$  is a component of the softmax output of the network.

### 3.3. Properties of CAS loss

The global minima for the uniformer term is any piecewise constant descriptor where each component of descriptor is constant on each individual region. This will make variance (uniformer term) of the descriptor zero on each region. The discriminator term eliminates the trivial minima of uniformer term where all descriptors on image are either zero or equal constant values. Hence the discriminator term is necessary to introduce the crucial discriminability properties to the learned descriptor. The discriminator term can be considered as an optimization problem of the form given below; the constraints below are due to the softmax layer at the end of the network.

Setting  $\alpha = 0$  in Equation 1 for a binary segmentation problem (with regions  $r_0$  and  $r_1$ ) leaves us with the task to,

$$\begin{aligned} &\text{maximize} \quad \|\hat{\mathbf{s}}(r_0) - \hat{\mathbf{s}}(r_1)\|_2^2 \\ &\text{subject to} \quad \sum_{m=1}^M \hat{s}^m(r_0) = 1 \quad \sum_{m=1}^M \hat{s}^m(r_1) = 1, \quad (3) \\ &\quad \hat{s}^m(r_0) \geq 0 \quad \hat{s}^m(r_1) \geq 0 \quad \forall m \end{aligned}$$

Some useful properties of the CAS loss are:

**Sparcity:** With the inequality constraints in Eq. 3 we get the feasible region of the objective. Because of the equality constraints this feasible region is bounded. In  $M$ -dimensions each inequality constraint will represent a half-space. With the intersection of these half-spaces we will get a convex polytope. In this case, the optimal solutions will occur at the corners of the convex polytope, which are sparse. Here one component of each  $\hat{\mathbf{s}}(r_0)$  and  $\hat{\mathbf{s}}(r_1)$  is 1 and all other components are 0, and the non-zero components for  $\hat{\mathbf{s}}(r_0)$  and  $\hat{\mathbf{s}}(r_1)$  lie at different indices. Thus, each unique region (or texture) will be represented by a sparse descriptor.

**Robustness to Class Imbalance:** Cross entropy loss based methods for general and salient object segmentation methods suffer from class imbalance where the larger class weighs the learning more [16]. In contrast, the CAS loss function is immune to class imbalance-induced training issues as all the terms in Eq. 1 used for calculation are normalised by class (region) size. Therefore, regardless of the size of salient objects (number of pixels they cover), all regions contribute equally to the loss function. Hence, even a small region will be represented well in the descriptor space. We didn't normalise for salient or non-salient class in CAS loss and achieved state-of-the-art results.

**Boundedness:** As a consequence of using softmax outputs in the last layer of neural network, the CAS loss func-

tion has a defined upper and lower bound. Since  $s^c \in [0, 1]^{w \times h}$  (where  $w$  and  $h$  are width and height of the output channel respectively), and we have all mean values in the loss, thus both the uniformer and discriminator terms are bounded by 1. Thus, the value of loss function lies in a bounded interval  $(\alpha N_i, -(1 - \alpha)N_i]$ , where  $N_i$  is the number of regions in an image  $i$ . A loss value of  $-(1 - \alpha)N_i$  indicates perfect segmentation of all samples in the training set.

## 4. Experimental Setup

This section describes the experimental setup for the experiments in this paper. The codes were setup in PyTorch [30] using Python3.7. The experiments were run on Nvidia Quadro RTX 6000 GPU and Intel Xeon 2.60GHz CPU<sup>1</sup>.

### 4.1. Architecture

The main aim of this work was to present a class-agnostic segmentation loss, regardless of the architecture. Therefore, we used standard available architectures as our models. We used the standard FCN-ResNet-101 and DeepLab-v3 architecture backbones with softmax layer as the output layer of the model.

### 4.2. Saliency Detection Experiments

**Datasets** For an extensive evaluation of our methods, we tested our models on 7 datasets: MSRA-B (5000 images) [34] dataset consists of single salient object in an image; DUTS [35] dataset has explicit training and testing sets which are DUTS-TR (10553 images) and DUTS-TE (5019 images) respectively; ECSSD (1000 images)[39] dataset has semantically meaningful and complex objects, and textures, containing salient objects of different sizes; PASCAL-S (850 images) [21] dataset contains natural images with cluttered backgrounds; HKU-IS (4447 images) [20] dataset has multiple disconnected salient objects, some touching the boundary; THUR15k (6232 images)[5] dataset contains random internet images and does not necessarily contain a salient object in every image and DUT-OMRON (5168 images) [40] dataset contains one or two complex salient objects. The datasets used for training were MSRA-B [34] (split 6:4 ratio for training and testing) and DUTS-TR [41].

**Pre-processing and Post-processing** All images were resized to  $256 \times 256$  and standardised to have mean 0 and unit variance. Standardisation ensures that all parts of the image share equal weights, otherwise the larger pixel values tend to dominate the weights of the neural network. After standardisation the distribution of pixels resembles a

Gaussian curve centred at zero, which helps in faster convergence of the neural network. The network has 2 sparse output channels, the correct saliency map  $S$ , and  $1 - S$ . We choose the channel for the salient object label by calculating the correlation of the output channels with the saliency map on the validation set and selecting the channel with maximum correlation value as saliency map. In post-processing, the continuous output saliency map  $S \in [0, 1]^{256 \times 256}$  of the neural network is thresholded using the popular method [4] to get a binary output map  $B$ , which is calculated as  $B(x, y) = 1$ , if  $S(x, y) > T$  else 0, if  $S(x, y) \leq T$  where,  $T = 2 \times \text{mean}(S)$  is the threshold, and  $x$  and  $y$  denotes the pixel positions on the map.

**Evaluation Metrics** The standard evaluation metrics used in salient object detection are  $F_\beta$ -score and Mean Absolute Error (MAE). The evaluation metrics are calculated on the ground-truth mask  $G \in \{0, 1\}^{w \times h}$  and the binary map  $B \in \{0, 1\}^{w \times h}$  extracted from saliency map  $S \in [0, 1]^{w \times h}$  (where  $w$  and  $h$  denote the width and height respectively).

$F_\beta$ -score is the weighted harmonic mean of precision and recall, with a non-negative weight  $\beta$ . The  $F_\beta$ -score is computed as,  $F_\beta = \frac{(1+\beta^2)\text{Precision} \times \text{Recall}}{\beta^2 \text{Precision} + \text{Recall}}$ . Setting  $\beta = 1$  provides the standard F-score formula. However, because the traditional  $F$ -score suffers from interpolation flaw, dependency flaw and equal-importance flaw [25] which results in unfair comparison, weighted  $F_\beta$  score is used [23]. Like all previous works, the value of  $\beta^2$  was set to 0.3.

To address the true negative saliency assignments (i.e., correctly marked as non-salient) and reward it, the MAE score is calculated as the average of absolute error between the saliency map  $S$  and the ground truth mask  $G$ .  $MAE = \frac{1}{w \times h} \sum_{x=1}^w \sum_{y=1}^h |S(x, y) - G(x, y)|$ , where  $x$  and  $y$  denotes the pixel positions on the map. It is desired to be as low as possible.

**Models** We used two standard models as the backbone for our experiments, *Model-CE* model is FCN-ResNet-101 and DeepLab-v3 architecture with cross entropy loss and *Model-CAS* model is FCN-ResNet-101 and DeepLab-v3 architecture with CAS loss. Hence, in total we get 4 models, 2 with backbone FCN-ResNet-101 and 2 with DeepLab-v3. For completeness of comparison with CE based methods, we define a class-agnostic version of the cross-entropy (CACE) loss function for binary segmentation as  $\min(-y_i \log p_i, -(1 - y_i) \log p_i)$ , where for  $i^{th}$  example,  $y_i$  is the true-class label and  $p_i$  is the predicted probability of belonging to class. Notice that as we increase the number of classes the arguments of the min function increases combinatorially in the definition of class-agnostic version of CE loss. In the low-fidelity data setting we train the *ResNet-CACE* model, which is FCN-ResNet-101 architecture with class-agnostic version of cross entropy loss. For these models the settings such as architecture, optimis-

<sup>1</sup>Code available at [https://github.com/sofmonk/class\\_agnostic\\_loss\\_saliency](https://github.com/sofmonk/class_agnostic_loss_saliency)

ers, activations functions and hyperparameters were same, a comparison of these models consequently results in comparison of the loss function’s performance.

## 5. Experiments

We conducted experiments for salient object detection (SOD) in two settings, low-fidelity data setting we have noisy labels for regions where and the probability of incorrect class labels is 0.5. We also applied our CAS loss to general segmentation problem to show the general nature of the loss function for any segmentation application.

### 5.1. Salient Object Detection (SOD)

**Low-fidelity data SOD** We implemented our class-agnostic segmentation loss framework for salient object detection with low-fidelity training data where region labels are noisy. We control the level of noise by controlling the amount of error for labels of the region. Notice that the noise is for the region label and not pixel wise label. We compared our methods against state of the art methods in low fidelity data case where the region labels are noisy with probability of incorrect region label of 0.5. We trained 3 state-of-the-art models PFAN [43], BAS-Net [31] and Pool-Net [22], and 4 of our models, ResNet-CE, ResNet-CAS, ResNet-CACE and DeepLab-CAS in this setting. The purpose of this experiment was to empirically prove the class-agnostic property of CAS loss i.e., even if class labels are not correct. Consequently, annotators do not need to assign class labels to the segments as long as labels for different regions are distinct. We show that even in very noisy region label case CAS loss works. We also show that the class-agnostic version of cross-entropy fails in this setting.

**High-fidelity data SOD** To empirically verify the segmentation ability of CAS loss against the state-of-the-art methods, we compared these methods in high-fidelity data setting. Here the training data included the images and ground truth masks with accurate class labels. Following the brief discussion in Section 4.2, in total the following 7 models, were trained until convergence: *ResNet-m-CE*, *ResNet-m-CAS*, *ResNet-d-CE*, *ResNet-d-CAS*, where *m* and *d* denote training on MSRA-B and DUTS-TR datasets respectively, *ResNet101-pre-CAS* model was pre-trained on MSRA-B using cross entropy loss and then trained on MSRA-B using CAS loss, *DeepLab-CE* and *DeepLab-CAS* models were trained on DUTS-TR dataset, pretrained on COCO dataset.

### 5.2. Multi-Region General Segmentation

To show the applicability of CAS loss across domains we have applied it to the task of multi-region general segmentation, where an input image is divided into a number of regions based on appearance where the number of regions is not known a-priori. We have tested on BSDS500

[1] and PascalVOC2012 [8] datasets. BSDS500 consists of 200 training and testing images each and 100 validation images for segmentation. PASCALVOC2012 consists of 1464 training and 1449 testing images. We compare against state of the art metric learning techniques to show the efficacy of the propose loss function

### 5.3. Texture Segmentation

Experiments were conducted on real-world binary segmentation dataset presented in [13]. This is one of the most challenging binary segmentation datasets [13] with huge intrinsic and extrinsic variability of textures. The dataset is divided into 128 training and 128 testing images. We tested our CAS loss and CE loss and showed that CAS outperforms CE loss by a significant margin. We trained using CAS and CE loss on multiple state-of-the-art architectures (ResNet and DeepLab-v3) and showed that CAS is more suited to general segmentation task compared to CE loss.

## 6. Results

**Salient Object Detection:** The quantitative results for salient object detection in low-fidelity training data setting are summarised in Tables 1 and a few qualitative samples are show in Figure 2. Since state-of-the-art methods are based on CE type loss which is highly reliant on fidelity of class labels, these methods fail completely in low-fidelity training data cases, with performance drops of around 50 % on most datasets. This essentially means that pixels are randomly labelled as salient or non-salient. Our CAS loss is immune to any performance degradation in low-fidelity training data (and trains the model in a class-agnostic manner). ResNet-CAS results in low-fidelity case degrades only slightly whereas for DeepLab-CAS the performance improves.

We also compare performance of our CAS loss with state-of-the-art methods in high-fidelity training data setting. The quantitative results are summarized in Table 2 and a few qualitative samples are shown in Figure 2. Models trained using CAS loss perform equally well to the state-of-the-art methods and at times superior to these models. Our models achieve state-of-the-art results on 5 out of the 7 datasets. No other state-of-the-art method performs that well consistently on that many datasets. For comparison we also tested on the state of the art in metric learning [7] using same architecture trained on MSRA-B dataset, the results are summarised in Table 2, our method being superior. Another aspect to note is that without any tweak in the FCN-ResNet101 or DeepLab-v3 architectures our models beat the state-of-the-art results. There is a good probability that with further tweak or using different networks we might beat the state-of-the-art with even larger margins. However, we would like to restate that the key contribution of this

Model	MSRA-B		DUTS-TE		ECSSD		PASCAL-S		HKU-IS		THUR15k		DUT-OMRON	
	$F_{\beta} \uparrow$	MAE $\downarrow$	$F_{\beta} \uparrow$	MAE	$F_{\beta} \uparrow$	MAE $\downarrow$	$F_{\beta} \uparrow$	MAE $\downarrow$	$F_{\beta} \uparrow$	MAE $\downarrow$	$F_{\beta} \uparrow$	MAE $\downarrow$	$F_{\beta} \uparrow$	MAE $\downarrow$
PFAN [43]	0.580	0.502	0.532	0.512	0.588	0.510	0.611	0.493	0.565	0.518	0.541	0.513	0.537	0.511
BAS-Net [31]	0.585	0.619	0.444	0.682	0.572	0.619	0.655	0.623	0.527	0.637	0.431	0.660	0.442	0.686
PoolNet [22]	0.603	0.502	0.566	0.501	0.597	0.503	0.648	0.480	0.582	0.503	0.517	0.505	0.526	0.498
ResNet-CE	0.691	0.140	0.427	0.191	0.625	0.178	0.592	0.206	0.633	0.160	0.670	0.150	0.666	0.147
ResNet-CACE	0.872	0.076	0.808	0.103	0.803	0.114	0.754	0.150	0.822	0.094	0.834	0.108	0.816	0.095
ResNet-CAS	0.920	0.038	0.836	0.077	0.837	0.085	0.773	0.126	0.856	0.067	0.865	0.080	0.846	0.069
DeepLab-CAS	0.937	0.038	0.868	0.064	0.875	0.068	0.810	0.110	0.896	0.050	0.893	0.070	0.871	0.058

**red** represents our best score value on the dataset; **blue** represents the second best score on the dataset

Table 1. Results on Low-fidelity training on MSRA-B data: Our method is superior

Model	MSRA-B		DUTS-TE		ECSSD		PASCAL-S		HKU-IS		THUR15k		DUT-OMRON	
	$F_{\beta} \uparrow$	MAE $\downarrow$	$F_{\beta} \uparrow$	MAE $\downarrow$	$F_{\beta} \uparrow$	MAE $\downarrow$	$F_{\beta} \uparrow$	MAE $\downarrow$	$F_{\beta} \uparrow$	MAE $\downarrow$	$F_{\beta} \uparrow$	MAE $\downarrow$	$F_{\beta} \uparrow$	MAE $\downarrow$
BAS-Net [31]	-	-	0.860	0.047	0.942	0.037	0.854	0.076	0.921	0.039	-	-	0.805	0.056
PoolNet [22]	-	-	0.892	0.036	0.945	0.038	0.880	0.065	0.935	0.030	-	-	0.833	0.053
CPSNet [42]	-	-	-	-	0.878	0.096	0.790	0.134	-	-	-	-	0.718	0.114
PFAN [43]	-	-	0.870	0.040	0.931	0.032	0.892	0.067	0.926	0.032	-	-	0.855	0.041
PAGENET+CRF[36]	-	-	0.817	0.047	0.926	0.035	0.835	0.074	0.920	0.030	-	-	0.770	0.063
PAGENET[36]	-	-	0.815	0.051	0.924	0.042	0.835	0.078	0.918	0.037	-	-	0.770	0.066
HED [11]	0.927	0.028	-	-	0.915	0.052	0.830	0.080	0.913	0.039	-	-	0.764	0.070
DNA [24]	-	-	0.873	0.040	0.938	0.040	-	-	0.934	0.029	0.796	0.068	0.805	0.056
GateNet [44]	-	-	0.898	0.035	0.952	0.035	0.888	0.065	0.943	0.029	-	-	0.829	0.051
LDF [37]	-	-	0.910	0.034	0.930	0.034	0.848	0.060	0.914	0.027	0.764	0.064	0.773	0.051
[29]	-	-	0.825	0.037	0.911	0.033	0.821	0.064	0.899	0.028	-	-	0.738	0.055
ITSD [45]	-	-	0.883	0.041	0.947	0.035	0.871	0.071	0.934	0.031	-	-	0.824	0.061
Discriminative loss [7]	0.905	0.052	0.829	0.082	0.829	0.093	0.756	0.134	0.840	0.077	0.846	0.086	0.834	0.072
ResNet-CAS (ours)	0.985 <sup>+</sup>	0.010 <sup>+</sup>	0.871 <sup>+</sup>	0.071 <sup>+</sup>	0.888 <sup>+</sup>	0.071 <sup>+</sup>	0.840 <sup>+</sup>	0.112 <sup>+</sup>	0.939 <sup>+</sup>	0.050 <sup>+</sup>	0.931 <sup>+</sup>	0.073 <sup>+</sup>	0.876 <sup>+</sup>	0.066 <sup>+</sup>
ResNet-CE (ours)	0.958 <sup>*</sup>	0.030 <sup>*</sup>	0.919 <sup>+</sup>	0.055 <sup>+</sup>	0.905 <sup>+</sup>	0.068 <sup>+</sup>	0.876 <sup>+</sup>	0.091 <sup>+</sup>	0.928 <sup>+</sup>	0.044 <sup>+</sup>	0.935 <sup>+</sup>	0.057 <sup>+</sup>	0.920 <sup>+</sup>	0.059 <sup>+</sup>
DeepLab-CAS (ours)	0.931	0.040	0.850	0.070	0.864	0.072	0.800	0.111	0.882	0.054	0.888	0.069	0.865	0.060
DeepLab-CE (ours)	0.928	0.039	0.847	0.070	0.867	0.069	0.805	0.110	0.880	0.052	0.881	0.070	0.856	0.061

\* represents model trained on MSRA-B dataset, + represents model trained on DUTS-TE dataset, ' represents model pre-trained on cross-entropy, red represents the best score value on the dataset, blue represents the second best score on the dataset, - represents the dataset was not tested by the method

Table 2. Numerical Results on High-fidelity data setting: Despite being designed for low-fidelity setting, our method performs equally well and at times superior to the state of the art method in high-fidelity setting as well.












































































	Image	Ground Truth	ResNet-pre-CAS	ResNet-m-CAS	ResNet-d-CAS	ResNet-m-CE	ResNet-d-CE	ResNet-CAS	ResNet-CE	ResNet-CACE	BAS-Net [31]	PoolNet [22]	CPSNet [42]	PFAN [43]	Low-fidelity trained sota
			High-fidelity trained Our Models					Low-fidelity trained Our Models			State-of-the-art				
1															
2															
3															
4															
5															

Figure 2. Visual Results for High-fidelity and Low-fidelity Data Training, and Comparison with State-of-the-art methods

work is the introduction of the CAS loss, and hence we did not tune the network architectures for superior results.

It is also worth mentioning that we performed the most basic level of distortion to the class labels to generate low-fidelity training data; even if the class label  $l$  was misla-

belled as *10* in some images, the CAS loss would still work. This is a consequence of dependence of CAS loss on  $s(r_i)$  and  $|r_i|$  but not the label of  $r_i$  (Equation 1).

**Multi-Object Segmentation** The results for multi-object segmentation are summarised in Table 5. Our loss



ResNet-CAS			ResNet-CE		
% of low-fidelity data	$F_\beta \uparrow$	MAE $\downarrow$	% of low-fidelity data	$F_\beta \uparrow$	MAE $\downarrow$
2	<b>0.903</b>	0.055	2	<b>0.903</b>	<b>0.051</b>
5	<b>0.921</b>	<b>0.046</b>	5	0.901	0.055
10	<b>0.904</b>	<b>0.053</b>	10	0.858	0.068
30	<b>0.906</b>	<b>0.520</b>	30	0.647	0.159
50	<b>0.920</b>	<b>0.038</b>	50	0.691	0.140

Table 3. Results: models trained on varying lfd MSRA-B data

	Region metrics		
	GT-cov. $\uparrow$ ODS	Rand. Index $\uparrow$ ODS	Var. Info. $\downarrow$ ODS
[7]	0.36	0.68	2.12
[32]	0.28	0.65	3.23
[33]	0.37	0.69	2.05
CAS (ours)	<b>0.38</b>	<b>0.71</b>	<b>1.98</b>

Table 4. Results for BSDS500 dataset [1]: Higher ground truth covering (GT-cov), and rand index, and lower variation of information (Var. Info) indicates a better fit to ground truth.

Method	[7]	[32]	[33]	CAS (ours)
Average Precision $\uparrow$	0.61	0.12	0.61	<b>0.71</b>

Table 5. Results for PASCAL VOC 2012 dataset [8]

	Contour		Region metrics					
	F-meas. $\uparrow$		GT-cov.		Rand. Index		Var. Info.	
	ODS	OIS	ODS	OIS	ODS	OIS	ODS	OIS
FCN-ResNet101-a-CE	0.04	0.04	0.79	0.79	0.55	0.55	1.39	1.39
FCN-ResNet101-a-CAS	<b>0.48</b>	<b>0.48</b>	<b>0.87</b>	<b>0.87</b>	<b>0.87</b>	<b>0.87</b>	<b>0.50</b>	<b>0.50</b>
FCN-ResNet101-t-CAS	0.17	0.17	0.83	0.83	0.66	0.66	0.94	0.94
DeepLab-d-CE	0.18	0.18	0.82	0.82	0.67	0.67	1.35	1.35
DeepLab-d-CAS	0.07	0.07	0.73	0.73	0.54	0.54	1.64	1.64

-a- denotes trained on the 7 saliency datasets; -t- denotes trained on texture data;  
-d- denotes trained on DUTS-TR data

Table 6. Results on Texture Segmentation Datasets of Deep Networks: Evaluated using contour and region metrics.

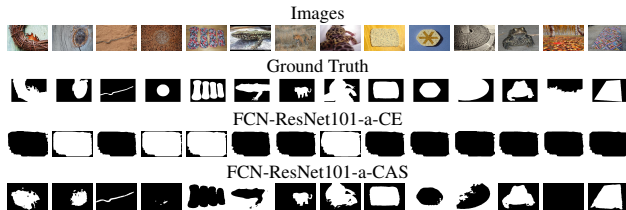


Figure 3. Sample representative results on Real-World Texture Dataset: Visual results for texture segmentation experiments; -a- denotes trained on the 7 saliency datasets

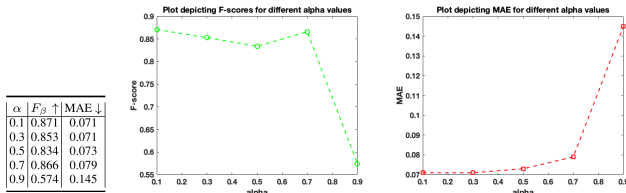


Table 7. Results different values of  $\alpha$  trained and tested on DUTS dataset with FCN-ResNet101

function out-performs state of the art metric learning methods by a significant margin.

**Texture Segmentation** Results for texture segmentation experiments are summarised in Table 6. Notice that CE

loss fails to learn any useful descriptor with both ResNet and DeepLab architectures because the real-world texture dataset is annotated with region segments and not region class labels. On the other hand, CAS loss outperforms CE loss and performs reasonably well in capturing complex texture segments (in majority of cases). Undeniably the results are not perfect since we have a very small texture segmentation training set at hand, nonetheless the point about the strength of CAS loss over CE loss is well demonstrated in this experiment.

**Ablation Study for Hyperparameter  $\alpha$**  The CAS loss (Equation 1) has one hyperparameter,  $\alpha$ , which sets the weights on the uniformer and discriminator terms. As argued in Section 3, the discriminator term has more important properties, consequently we weighed this term more.  $\alpha$  was set to 0.1. This was validated by testing for various values of  $\alpha$  as shown in Table 7. The learning rate was 1e-3 with Adam [16] optimizer.

## 7. Conclusion

We presented class-agnostic segmentation loss function which allows us to cast the problem of region-based general segmentation problem with deep networks. We tested on 7 salient object segmentation datasets against 15 methods (in HFD and LFD settings) and on challenging multi-object general segmentation dataset and a texture dataset. Using the class-agnostic segmentation loss function we tackled the problem of salient object segmentation in low-fidelity training data case and showed state-of-the-art results, around 50% better than the next best methods. This huge performance gain is due to the fact that CAS loss forces learning of descriptors through the deep network which are invariant on similar looking regions, which also shows in the high-fidelity setting. We also applied CAS loss to high-fidelity training data case as well as texture segmentation and multi-object segmentation and showed state-of-the-art results. This shows that our CAS loss performs well in multiple scenarios for general segmentation across domains.

Although standard CE based loss functions perform satisfactorily in salient object detection with high-fidelity training data, they fail completely in low-fidelity training data case with a performance drop of around 50%. Likewise the CE based loss functions fail to learn any significant feature in texture segmentation dataset, and CAS outperforms competing metric learning based methods on multi-object segmentation task. Notice, that we have used saliency detection and general segmentation as applications for our class-agnostic segmentation loss. However, the utility of our loss is not restricted to these applications and it can be applied to any general segmentation problem.



## References

- [1] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. 33(5):898–916.
- [2] Liang Chieh Chen, Jonathan T. Barron, George Papandreou, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with task-specific edge detection using CNNs and a discriminatively trained domain transform. Technical report, 2016.
- [3] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking Atrous Convolution for Semantic Image Segmentation. jun 2017.
- [4] M.M. Cheng, G.X. Zhang, N.J. Mitra, Xiaolei Huang, and S.M. Hu. Global contrast based salient region detection. In *CVPR*, pages 409–416, 2011.
- [5] Ming Ming Cheng, Niloy J Mitra, Xiaolei Huang, and Shi Min Hu. SalientShape: Group saliency in image collections. *Visual Computer*, 30(4):443–453, 2014.
- [6] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. Technical report, 2005.
- [7] Bert De Brabandere, Davy Neven, and Luc Van Gool. Semantic Instance Segmentation with a Discriminative Loss Function. Technical report, 2017.
- [8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2011 (VOC2011) Results. <http://www.pascal-network.org/challenges/VOC/voc2011/workshop/index.html>.
- [9] Alireza Fathi, Zbigniew Wojna, Vivek Rathod, Peng Wang, Hyun Oh Song, Sergio Guadarrama, and Kevin P. Murphy. Semantic Instance Segmentation via Deep Metric Learning. 2017.
- [10] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Networks. jun 2014.
- [11] Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip Torr. Deeply supervised salient object detection with short connections, 2017.
- [12] Jia Deng, Wei Dong, R. Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. pages 248–255, 2009.
- [13] Naeemullah Khan, Marei Algarni, Anthony Yezzi, and Ganesh Sundaramoorthi. Shape-tailored local descriptors and their application to segmentation and tracking. In *CVPR*, volume 07-12-June, pages 3890–3899, 2015.
- [14] Naeemullah Khan, Byung-Woo Hong, Anthony Yezzi, and Ganesh Sundaramoorthi. Coarse-to-Fine Segmentation with Shape-Tailored Continuum Scale Spaces. In *CVPR*, pages 1733–1742. IEEE, jul 2017.
- [15] Naeemullah Khan and Ganesh Sundaramoorthi. Learned Shape-Tailored Descriptors for Segmentation. In *CVPR*, pages 666–674. IEEE Computer Society, dec 2018.
- [16] Diederik P Kingma and Jimmy Lei Ba. ADAM: A Method for Stochastic Optimization. Technical report, 2014.
- [17] Iasonas Kokkinos. Pushing the Boundaries of Boundary Detection using Deep Learning. Technical report, 2015.
- [18] Shu Kong and Charless Fowlkes. Recurrent Pixel Embedding for Instance Grouping. In *CVPR*, pages 9018–9028, 2018.
- [19] Srinivas S S Kruthiventi, Kumar Ayush, and R Venkatesh Babu. DeepFix: A Fully Convolutional Neural Network for Predicting Human Eye Fixations. *IEEE Transactions on Image Processing*, 26(9):4446–4456, 2017.
- [20] Guanbin Li and Yizhou Yu. Visual saliency detection based on multiscale deep CNN features. Technical Report 11, 2016.
- [21] Yin Li, Xiaodi Hou, Christof Koch, James M Rehg, and Alan L Yuille. The secrets of salient object segmentation. In *CVPR*, pages 280–287, 2014.
- [22] Jiang-Jiang Liu, Qibin Hou, Ming-Ming Cheng, Jiashi Feng, and Jianmin Jiang. A Simple Pooling-Based Design for Real-Time Saliency Object Detection. In *CVPR*, 2019.
- [23] Nian Liu, Junwei Han, and Ming Hsuan Yang. PiCANet: Learning Pixel-Wise Contextual Attention for Saliency Detection. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3089–3098, 2018.
- [24] Yun Liu, Deng-Ping Fan, Guang-Yu Nie, Xinyu Zhang, Vahan Petrosyan, and Ming-Ming Cheng. DNA: Deeply-supervised Nonlinear Aggregation for Saliency Object Detection. 2019.
- [25] Ran Margolin, Lihi Zelnik-Manor, and Ayellet Tal. How to evaluate foreground maps. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2014.
- [26] Davy Neven, Bert De Brabandere, Marc Proesmans, and Luc Van Gool. Instance segmentation by jointly optimizing spatial embeddings and clustering bandwidth. Technical report, 2019.
- [27] Junting Pan, Cristian Canton Ferrer, Kevin McGuinness, Noel E. O’Connor, Jordi Torres, Elisa Sayrol, and Xavier Giro-i Nieto. SalGAN: Visual Saliency Prediction with Generative Adversarial Networks. 2017.
- [28] Junting Pan, Elisa Sayrol, Xavier Giro-I-Nieto, Kevin McGuinness, and Noel E O’connor. Shallow and deep convolutional networks for saliency prediction. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2016-Decem, pages 598–606, 2016.
- [29] Youwei Pang, Xiaoqi Zhao, Lihe Zhang, and Huchuan Lu. Multi-Scale Interactive Network for Saliency Object Detection. In *CVPR 2020*, pages 9413–9422, 2020.
- [30] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. Technical report, 2017.
- [31] Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jagersand. BASNet: Boundary-Aware Saliency Object Detection. In *CVPR*, 2019.
- [32] Oren Rippel, Manohar Paluri, Piotr Dollar, and Lubomir Bourdev. Metric Learning with Adaptive Density Discrimination. 2016. arXiv: 1511.05939.

- [33] Florian Schroff, Dmitry Kalenichenko, and James Philbin. FaceNet: A unified embedding for face recognition and clustering. In *CVPR*, volume 07-12-June, pages 815–823, 2015.
- [34] Jingdong Wang, Huaizu Jiang, Zejian Yuan, Ming-Ming Cheng, Xiaowei Hu, and Nanning Zheng. Salient object detection: A discriminative regional feature integration approach. *International Journal of Computer Vision*, 123(2):251–268, 2017.
- [35] Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan. Learning to detect salient objects with image-level supervision. In *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, volume 2017-Janua, pages 3796–3805. IEEE, jul 2017.
- [36] Wenguan Wang, Shuyang Zhao, Jianbing Shen, Steven C H Hoi, and Ali Borji. Salient Object Detection with Pyramid Attention and Salient Edges. *CVPR*, 1(c):1448–1457, 2019.
- [37] Jun Wei, Shuhui Wang, Zhe Wu, Chi Su, Qingming Huang, and Qi Tian. Label Decoupling Framework for Salient Object Detection. 2020. arXiv: 2008.11048.
- [38] Saining Xie and Zhuowen Tu. Holistically-Nested Edge Detection. In *ICCV*, 2015.
- [39] Qiong Yan, Li Xu, Jianping Shi, and Jiaya Jia. Hierarchical saliency detection. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1155–1162, 2013.
- [40] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming Hsuan Yang. Saliency detection via graph-based manifold ranking. Technical report, 2013.
- [41] Yu Zeng, Huchuan Lu, Lihe Zhang, Mengyang Feng, and Ali Borji. Learning to Promote Saliency Detectors. In *CVPR*, pages 1–7, 2018.
- [42] Yu Zeng, Yunzhi Zhuge, Huchuan Lu, Lihe Zhang, Mingyang Qian, and Yizhou Yu. Multi-source weak supervision for saliency detection. *CVPR*, 2019.
- [43] Ting Zhao and Xiangqian Wu. Pyramid Feature Attention Network for Saliency detection. *CVPR*, 2019.
- [44] Xiaoqi Zhao, Youwei Pang, Lihe Zhang, Huchuan Lu, and Lei Zhang. Suppress and Balance: A Simple Gated Network for Salient Object Detection. In *ECCV*, 2020. arXiv: 2007.08074.
- [45] Huajun Zhou, Xiaohua Xie, Jian-Huang Lai, Zixuan Chen, and Lingxiao Yang. Interactive Two-Stream Decoder for Accurate and Fast Saliency Detection. In *CVPR 2020*, pages 9141–9150, 2020.