Optimizing the F-measure for Threshold-free Salient Object Detection

Kai Zhao
Nankai University
kz@mail.nankai.edu.cn

Shanghua Gao
Nankai University
shanghuagao@gmail.com

Qibin Hou
Nankai University
andrewhoux@gmail.com

Dan-dan Li
Shanghai University
dandanli814@gmail.com

Ming-Ming Cheng
Nankai University
cmm@nankai.edu.cn

Abstract

Current CNN-based solutions to salient object detection (SOD) mainly rely on the optimization of cross-entropy loss (CELoss). Then the quality of detected saliency maps is often evaluated in terms of F-measure. In this paper, we investigate an interesting issue: can we consistently use the F-measure formulation in both training and evaluation for SOD? By reformulating the standard F-measure we propose the *relaxed F-measure* which is differentiable w.r.t the posterior and can be easily appended to the back of CNNs as the loss function. Compared to the conventional cross-entropy loss of which the gradients decrease dramatically in the saturated area, our loss function, named FLoss, holds considerable gradients even when the activation approaches the target. Consequently, the FLoss can continuously force the network to produce polarized activations. Comprehensive benchmarks on several popular datasets show that FLoss outperforms the state-of-the-arts with a considerable margin. More specifically, due to the polarized predictions, our method is able to obtain high quality saliency maps without carefully tuning the optimal threshold, showing significant advantages in real world applications.

1 Introduction

We consider the task of salient object detection (SOD), where each pixel of a given image has to be classified as salient (outstanding) or not. Human visual system is able to perceive and process visual signals distinctively: interested regions are conceived and analyzed with high priority while other regions draw less attention. This capacity has been long studied in the computer vision community in the name of 'salient object detection', since it can ease the procedure of scene understanding [2]. The performance of modern salient object detection methods is often evaluated in terms of F-measure. Rooted from information retrieval [22], the F-measure is widely used as an evaluation metric in tasks where elements of a specified class have to be retrieved, especially when the relevant class is rare. Given the per-pixel prediction $\hat{Y}(\hat{y}_i \in [0,1], i = 1, ..., |Y|)$ and the ground-truth saliency map $Y(y_i \in \{0,1\}, i = 1, ..., |Y|)$, a threshold t is applied to obtain the binarized prediction $\dot{Y}^t(\dot{y}_i^t \in \{0,1\}, i = 1, ..., |Y|)$. The F-measure is then defined as the harmonic mean of precision and recall:

$$F(Y, \dot{Y}^t) = (1 + \beta^2) \frac{\text{precision}(Y, \dot{Y}^t) \cdot \text{recall}(Y, \dot{Y}^t)}{\beta^2 \text{precision}(Y, \dot{Y}^t) + \text{recall}(Y, \dot{Y}^t)} \tag{1}$$

where $\beta^2 > 0$ is a balance factor between precision and recall. When $\beta^2 > 1$ the F-measure is biased in favour of recall and otherwise the F-measure considers precision more than recall.

Preprint. Work in progress.

Most CNN-based solutions for SOD [7, 11, 23] mainly rely on the optimization of cross-entropy loss (CELoss) in a FCN [15] architecture, and the quality of saliency maps is often assessed by the F-measure. Optimizing the pixel-independent CELoss can be regarded as minimizing the mean absolute error (MAE=$\frac{\sum_i^N |\hat{y}_i - y_i|}{N}$), because in both circumstances each prediction/ground-truth pair works independently and contributes to the final score equally. Models trained with CELoss, if the data labels have biased distribution, would make biased predictions towards the majority class. Therefore, salient object detectors trained with CELoss hold biased prior and tend to predict unknown pixels as the background, consequently leading to low-recall detections. The F-measure [22] is a more sophisticated and comprehensive evaluation metric which combines precision and recall into a single score and automatically offsets the unbalance between positive/negative samples.

In this paper, we investigate to provide a uniform formulation in both training and evaluation for salient object detection. By directly training with the evaluation metric, *i.e.* the F-measure, as the optimization target, we perform F-measure maximizing in an end-to-end manner. To perform the end-to-end learning, we propose the *relaxed F-measure* to overcome the undifferentiability in the standard F-measure formulation. The proposed loss function, named FLoss, is decomposable w.r.t the posterior $\hat{Y}$ and thus can be appended to the back of a CNN as supervision without effort. We test the FLoss on several state-of-the-art SOD architectures and witness a visible performance gain. Furthermore, the proposed FLoss holds considerable gradients even in the saturated area, resulting in polarized predictions that are stable against the threshold.

Our proposed FLoss enjoys following favorable properties:

- Threshold-free salient object detection. Models trained with FLoss produce contrastive saliency maps in which the foreground and background are clearly separated, therefore, FLoss can achieve high performance regardless of the threshold. Although post-processing technologies like CRF can be used to obtain contrastive salient maps as well, however, these methods are often time-consuming. For instance, in [7] the CRF takes about 400ms on a 300x400 image, while the salient object detection part itself only costs 80ms.
- Being able to deal with unbalanced data. Defined as the harmonic mean of precision and recall, the F-measure is able to establish a balance between samples of different classes. We experimentally evidence that our method can find a better compromise between precision and recall.

2 Related Work

We review several CNN-based architectures for salient object detection and the literature related to F-measure optimization.

Salient Object Detection (SOD): The convolutional neural network (CNN) is proven to be dominant in many sub-areas of computer vision. Significant progress has been achieved since the presence of CNN in salient object detection. The DHS net [13] is one of the pioneers of using CNN for salient object detection. DHS firstly produces a coarse saliency map with global cues including contrast, objectness *et al.*, then the coarse map is progressively refined with a hierarchical recurrent convolutional neural network. The emergence of the fully convolutional network (FCN) [15] provides an elegant way to perform the end-to-end pixel-wise inference. DCL [11] uses a two-stream architecture to process contrast information in both pixel and patch levels. The FCN-based sub-stream produces a saliency map with pixel-wise accuracy, and the other network stream performs inference on each object segment. Finally, a fully connected CRF [9] is used to combine the pixel-level and segment-level semantics.

Rooted from the HED [24] for edge detection, aggregating multi-scale side-outputs is proven to be effective in refining dense predictions especially when the detailed local structures are required to be preserved. In the HED-like architectures, deeper side-outputs capture rich semantics and shallower side-outputs contain high-resolution details. Combining these representations of different levels will lead to significant performance improvements. DSS [7] introduces deep-to-shallow short connections across different side-outputs to refine the shallow side-outputs with deep semantic features. The deep-to-shallow short connections enable the shallow side-outputs to distinguish real salient objects from the background, and meanwhile retain the high resolution. The similar idea has been adopted by Zhang *et al.* in Amulet [26].

These methods mentioned above tried to refine salient object detection by introducing a more powerful network architecture, from recurrent refining network to multi-scale side-output fusing.

F-measure Optimization: Despite having been utilized as a common performance metric in many application domains, optimizing the F-measure doesn't draw much attention until very recently. The works aiming at optimizing the F-measure can be divided into two subcategories [4]: (a) structured loss minimization methods such as [19, 20] which optimize the F-measure as the target during training; and (b) plug-in rule approaches which optimize the F-measure during inference phase [5, 8, 18, 21].

Much of the attention has been drawn to the study of the latter subcategory: finding an optimal threshold value which leads to a maximal F-measure given predicted posterior $\hat{Y}$. There are few articles about optimizing the F-measure during the training phase. Petterson *et al.* [19] optimize the F-measure indirectly by maximizing a loss function associated to the F-measure. Then in their successive work [20] they construct an upper bound of the discrete F-measure, and then maximize the F-measure by optimizing its upper bound.

3 Optimizing the F-measure in CNNs

3.1 The Relaxed F-measure

In the standard F-measure, the true positive, false positive and false negative are defined as the number of corresponding samples:

$$\begin{aligned} TP(\dot{Y}^t, Y) &= \sum_i 1(y_i == 1 \text{ and } \dot{y}_i^t == 1) \\ FP(\dot{Y}^t, Y) &= \sum_i 1(y_i == 0 \text{ and } \dot{y}_i^t == 1) \\ FN(\dot{Y}^t, Y) &= \sum_i 1(y_i == 1 \text{ and } \dot{y}_i^t == 0) \end{aligned} \tag{2}$$

where Y is the ground-truth, $\dot{Y}^t$ is the binary prediction binarized by threshold t and Y is the ground-truth saliency map. $1(\cdot)$ is an indicator function that evaluates to 1 if its argument is true and 0 otherwise.

To incorporate the F-measure into CNN and optimize in an end-to-end manner, we have to define a decomposable F-measure that is differentiable over prediction $\hat{y}$. Based on this motivation, we reformulate the true positive, false positive and false negative based on the posterior $\hat{Y}$:

$$\begin{aligned} TP(\hat{Y}, Y) &= \sum_i \hat{y}_i \cdot y_i \\ FP(\hat{Y}, Y) &= \sum_i \hat{y}_i \cdot (1 - y_i) \\ FN(\hat{Y}, Y) &= \sum_i (1 - \hat{y}_i) \cdot y_i \end{aligned} \tag{3}$$

Similar formulation has been used in [16] to evaluate the quality of saliency maps, which is proven to be consistent with human perception. Given the definitions in Eq.3, precision p and recall r are:

$$p(\hat{Y}, Y) = \frac{TP}{TP + FP}, \quad r(\hat{Y}, Y) = \frac{TP}{TP + FN} \tag{4}$$

Finally the *relaxed F-measure* is defined as below:

$$\begin{aligned} F(\hat{Y}, Y) &= \frac{(1 + \beta^2) p \cdot r}{\beta^2 p + r} \\ &= \frac{(1 + \beta^2) TP}{\beta^2 (TP + FN) + (TP + FP)} \\ &= \frac{(1 + \beta^2) TP}{H} \end{aligned} \tag{5}$$

where $H = \beta^2 (TP + FN) + (TP + FP)$.

3.2 Maximizing F-measure in CNNs

In order to maximize the *relaxed F-measure* in CNNs in an end-to-end manner, we define our proposed F-measure based loss (FLoss) function $\mathcal{L}_F$ as:

$$\mathcal{L}_F(\hat{Y}, Y) = 1 - F = 1 - \frac{(1+\beta^2)TP}{H} \tag{6}$$

Minimizing $\mathcal{L}_F(\hat{Y}, Y)$ is equivalent to maximizing the *relaxed F-measure*. Note that $\mathcal{L}_F$ is calculated directly from the raw prediction $\hat{Y}$ without thresholding. Therefore, $\mathcal{L}_F$ is differentiable over the prediction $\hat{Y}$ and can be plugged into CNNs without effort. The partial derivative of loss $\mathcal{L}_F$ over network activation $\hat{Y}$ at location i is:

$$\begin{aligned}
\frac{\partial \mathcal{L}_F}{\partial \hat{y}_i} &= -\frac{\partial F}{\partial \hat{y}_i} \\
&= -\Big(\frac{\partial F}{\partial TP} \cdot \frac{\partial TP}{\partial \hat{y}_i} + \frac{\partial F}{\partial H} \cdot \frac{\partial H}{\partial \hat{y}_i}\Big) \\
&= -\Big(\frac{(1+\beta^2)y_i}{H} - \frac{(1+\beta^2)TP}{H^2}\Big) \\
&= \frac{(1+\beta^2)TP}{H^2} - \frac{(1+\beta^2)y_i}{H}
\end{aligned} \tag{7}$$

3.3 FLoss vs Cross-entropy Loss

We compare the proposed FLoss with widely used pixel-wise cross-entropy loss (CELoss) which is defined as:

$$\mathcal{L}_{CE}(\hat{Y}, Y) = -\sum_{i}^{|Y|} y_i \log \hat{y}_i + (1-y_i)\log(1-\hat{y}_i) \tag{8}$$

where i is the spatial location of the input image and $|Y|$ is the number of pixels of the input image. The gradient of $\mathcal{L}_{CE}$ w.r.t prediction $\hat{y}_i$ is:

$$\frac{\partial \mathcal{L}_{CE}}{\partial \hat{y}_i} = \frac{y_i}{\hat{y}_i} - \frac{1-y_i}{1-\hat{y}_i} \tag{9}$$

As revealed in Eq. 7 and Eq. 9, the gradient of CELoss $\frac{\partial \mathcal{L}_{CE}}{\partial \hat{y}_i}$ relies only on the prediction/ground-truth of a single pixel i; whereas in FLoss $\frac{\partial \mathcal{L}_F}{\partial \hat{y}_i}$ is globally determined by the prediction and ground-truth of ALL pixels in the image.

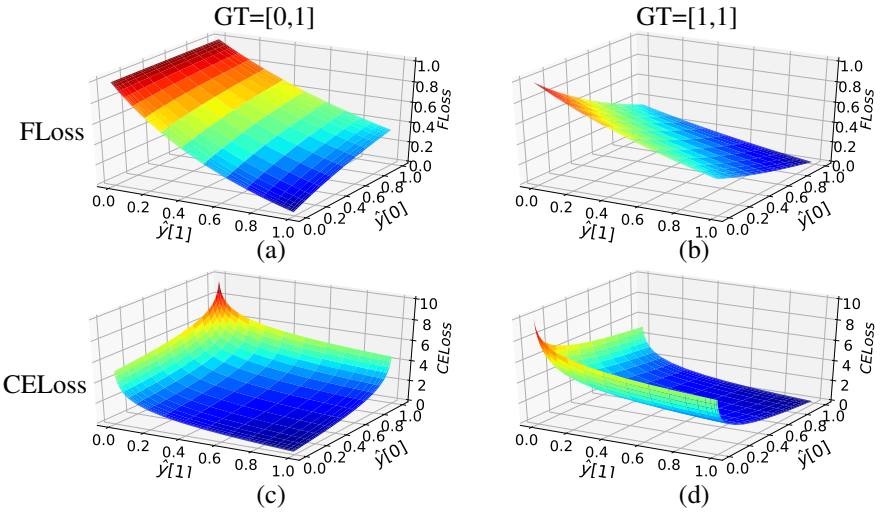


Figure 1: Surface plots of FLoss and CELoss in the case of 2 points binary classification problem.

We further compare the surface plots (Fig. 1) of FLoss and CELoss in a two points binary classification problem under different ground-truth settings (GT=[0, 1] and GT=[1, 1]). The two spatial axes represent the prediction $\hat{y}_0$ and $\hat{y}_1$, and the z axis is the loss.

As shown in Fig. 1, the gradient of FLoss is different from that of CELoss in two aspects: (1) Limited gradient: the FLoss holds limited gradient values even the predictions are far away from the ground-truth. This is crucial for CNN training because it prevents the notorious gradient explosion problem. Additionally, it allows a larger learning rate in the training phase, as evidenced by the experiments. (2) Considerable gradients in the saturated area: in CELoss, the gradient decays when the prediction gets closer to the ground-truth, while FLoss holds considerable gradients even in the saturated area. This will force the network to have polarized predictions. Salient detection examples in Fig. 2 illustrate the 'high contrast' and polarized predictions.

4 Experiments and Analysis

4.1 Experimental Configurations

Dataset and data augmentation. We uniformly train our model and competitors on the MSRA-B [14] training set for a fair comparison. The MSRA-B dataset with 5000 images in total is equally split into training/testing subsets. We test the trained models on 5 other SOD datasets: ECSSD [25], HKU-IS [10], PASCALS [12], SOD [17], and DUT-OMRON [17]. More statistics of these datasets are shown in Tab. 1. It's worth mentioning that the challenging degree of a dataset is determined by many factors such as the number of images, the number of objects in one image, the contrast of salient objects w.r.t the background, the complexity of salient object structures, the center bias of salient objects and the size variance of images *etc*. Analysing these details is out of the scope of this paper, we refer the readers to [6] for more analysis of datasets.

Data augmentation is critical to generating sufficient data for training deep CNNs. We fairly perform data augmentation for the original implementations and their FLoss variants. For the DSS [7] and DHS [13] architectures we perform only horizontal flip on both training images and saliency maps just as DSS did. Amulet [26] only allows 256×256 inputs. We randomly crop/pad the original data to get square images, then resize them to meet the shape requirement.

Dataset	#Images	Contrast
MSRA-B [14]	5000	High
ECSSD [25]	1000	High
HKU-IS [10]	4000	Low
PASCALS [12]	850	Medium
SOD [17]	300	Low
DUT-OMRON [17]	5168	Low

Table 1: Statistics of relevant SOD datasets. '#Images' indicates the number of images in a dataset.

Network architecture and hyper-parameters. We test our proposed FLoss on 3 baseline methods: Amulet [26], DHS [14] and DSS [7]. To verify the effectiveness of FLoss (Eq.6), we replace the cross-entropy loss (CELoss) layer(s) in the original implementations and keep all other network structures unchanged. As explained in Sec.3.3, the FLoss allows a larger base learning rate due to limited gradients. We use the base learning rate 10^4 times the original settings. For example, in DSS the base learning rate is 10^{-8}, while in our F-DSS, the base learning rate is 10^{-4}. All other hyper-parameters are consistent with the original implementations for a fair comparison.

Performance evaluation. We evaluate the performance of saliency maps in terms of maximal F-measure (MaxF), mean F-measure (MeanF) and mean absolute error (MAE = $\frac{\sum_i^N |\hat{y}_i - y_i|}{N}$). The factor β^2 in Eq. 1 is set to 0.3 as suggested by [1, 7, 11, 13, 23]. By applying series thresholds $t \in \mathcal{T}$ to the saliency map $\hat{Y}$, we obtain binarized saliency maps $\dot{Y}^t$ with different precisions, recalls and F-measures. Then the optimal threshold t_o and MaxF are achieved by tuning the threshold over the whole dataset:

$$\begin{aligned} t_o &= \underset{t \in \mathcal{T}}{\operatorname{argmax}} F(Y, \dot{Y}^t) \\ \text{MaxF} &= F(Y, \dot{Y}^{t_o}) \end{aligned} \tag{10}$$

The MeanF is the average F-measure of saliency maps under different thresholds:

$$\text{MeanF} = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} F(Y, \dot{Y}^t) \tag{11}$$

4.2 Detection Results Comparisons

We compare the proposed method with several baselines on 5 popular datasets. Some example detection results are shown in Fig. 2 and comprehensive quantitative comparisons are in Tab. 2. In general, methods with FLoss can obtain considerable improvements compared with their cross-entropy loss (CELoss) based counterparts especially in terms of MeanF and MAE which witnesses a nearly 2% performance gain on average. This is mainly because our method is stable against the threshold, leading to high-performance saliency maps under a wide threshold range. In our detected saliency maps, the foreground (salient objects) and background are well separated, as shown in Fig. 2 and explained in Sec. 3.3.

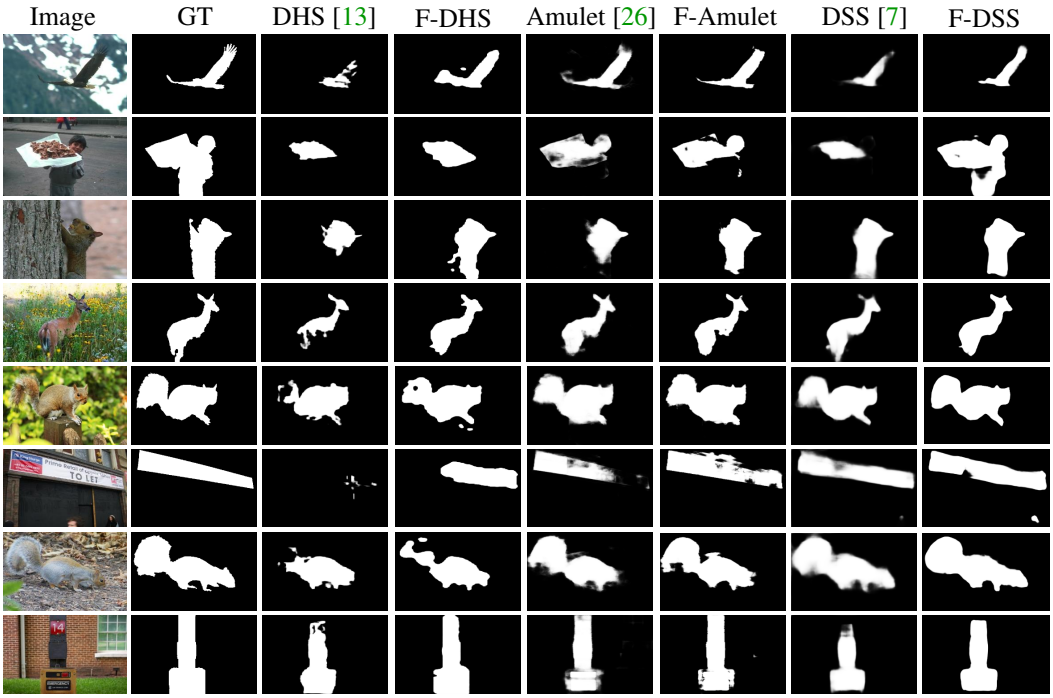


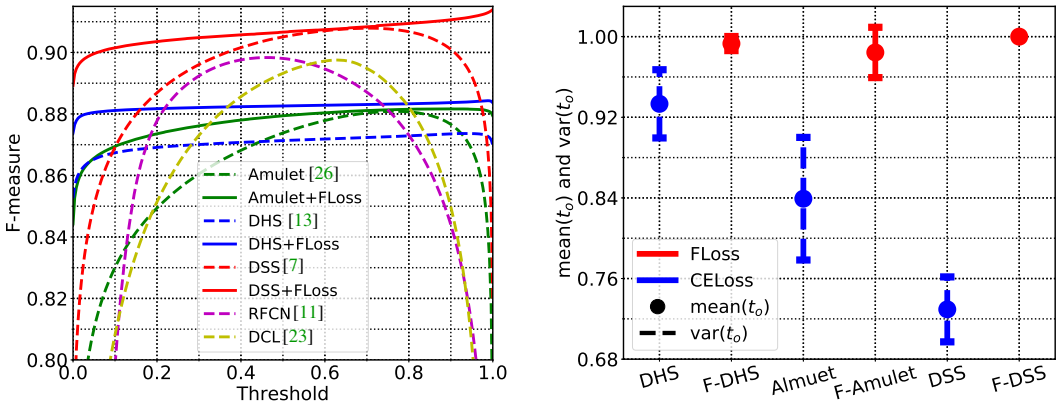
Figure 2: Salient object detection examples on several popular datasets. F-DHS, F-Amulet and F-DSS indicate the original architectures trained with our proposed FLoss. FLoss leads to sharp salient confidence especially on the object boundaries.

	Training data		ECSSD [25]			HKU-IS [10]			PASCALS [12]			SOD [17]			DUT-OMRON [17]		
Model	Train	#Images	MaxF	MeanF	MAE	MaxF	MeanF	MAE	MaxF	MeanF	MAE	MaxF	MeanF	MAE	MaxF	MeanF	MAE
RFCN [23]	MK [3]	10K	0.898	0.842	0.095	0.895	0.830	0.078	0.829	0.784	0.118	0.807	0.748	0.161	-	-	-
DCL [11]	MB [14]	2.5K	0.897	0.847	0.077	0.893	0.837	0.063	0.807	0.761	0.115	0.833	0.780	0.131	0.733	0.690	0.095
DHS [13]	MK [3]+D [17]	9.5K	0.905	0.876	0.066	0.891	0.860	0.059	0.820	0.794	0.101	0.819	0.793	0.136	-	-	-
Amulet [26]	MK [3]	10K	0.912	0.898	0.059	0.889	0.873	0.052	0.828	0.813	0.092	0.801	0.780	0.146	0.737	0.719	0.083
DHS [13]	MB [14]	2.5K	0.874	0.867	0.074	0.835	0.829	0.071	0.782	0.777	0.114	0.800	0.789	0.140	0.704	0.696	0.078
DHS+FLoss [13]	MB [14]	2.5K	0.884	0.879	0.067	0.859	0.854	0.061	0.792	0.786	0.107	0.801	0.795	0.138	0.707	0.701	0.079
Amulet [26]	MB [14]	2.5K	0.881	0.857	0.076	0.868	0.837	0.061	0.775	0.753	0.125	0.791	0.776	0.149	0.704	0.663	0.098
Amulet-FLoss	MB [14]	2.5K	0.894	0.883	0.063	0.880	0.866	0.051	0.791	0.776	0.115	0.805	0.800	0.138	0.729	0.696	0.097
DSS [7]	MB [14]	2.5K	0.908	0.889	0.060	0.899	0.877	0.048	0.824	0.806	0.099	0.835	0.815	0.125	0.761	0.738	0.071
DSS+FLoss	MB [14]	2.5K	0.914	0.903	0.050	0.908	0.896	0.038	0.829	0.818	0.091	0.843	0.838	0.111	0.777	0.755	0.067

Table 2: Quantitative comparison of different methods on 5 popular datasets.

4.3 Threshold-free Salient Object Detection

We analyse the influences of thresholds in two aspects: (1) performance under different thresholds, which reflects the stability of a method against threshold change, and (2) mean and variance of t_o on different datasets, which represent the generalization abilities.



(a) (b)

Figure 3: (a) F-measures under different thresholds on the ECSSD dataset. (b) The mean and variance of optimal threshold t_o. FLoss-based methods hold stable t_o across different datasets (lower t_o variances) and different backbone architectures (F-DHS, F-Amulet and F-DSS hold very close mean t_o).

Fig.3 (a) illustrates the F-measure w.r.t different thresholds on the ECSSD dataset. For most methods without FLoss, the F-measure changes sharply with the threshold and the maximal F-measure (MaxF) presents only in a narrow threshold span, while architectures with FLoss are almost immune from the change of threshold. Fig.3 (b) reflects the mean and variance of t_o across different datasets. Methods without FLoss (DHS, DSS, Amulet) present more diverse t_o on different datasets, as evidenced by their large variances. While FLoss-based methods (F-DHS, F-Amulet, F-DSS) have more concentrated optimal thresholds on different datasets, as evidenced by their small var(t_o). Moreover, even with different backbone architectures (F-DHS, F-Amulet, F-DSS), FLoss still presents a stable optimal threshold. In conclusion, the performance of FLoss is stable against threshold (Fig.3 (a)), and the t_o of FLoss is stable across different datasets, regardless of what backbone architecture is used (Fig.3 (b)).

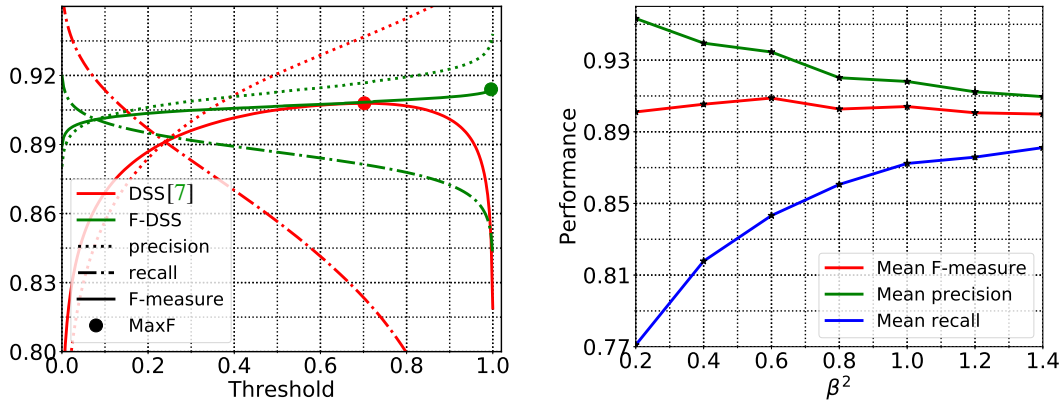
The standard evaluation protocol for salient object detection works as below: (a) a saliency map $\hat{Y}$ is obtained with the trained model; (b) a series of thresholds t are applied to the saliency maps, deriving binary saliency maps $\dot{Y}^t$; (d) tune the optimal threshold t_o and maximal F-measure (MaxF) using Eq. 10.

There is an obvious limitation in above procedure: the F-measure is sensitive to thresholds, as shown in Fig.3 (a), and there is no ground-truth in real-world circumstances to search such 'optimal threshold'. One solution is to tune the t_o on a validating set and then generalize to other images. However, as shown in Fig.3 (b), for most of the conventional methods the t_o varies among datasets, making it difficult to transfer t_o across different data. An 'optimal threshold' on one dataset may probably be sub-optimal on other datasets. Our proposed FLoss holds stable t_o on multiple datasets and backbone network architectures, showing great potential in real-world applications.

4.4 The Label-unbalancing Problem in SOD

The foreground and background are biased in SOD where most of the pixels belong to the non-salient background. The unbalanced training data will lead the model to local minimal that tends to predict unknown pixels as the background. Consequently, in evaluation recall becomes a bottleneck to the performance, as illustrated in Fig. 4 (a).

Although assigning loss weight to the positive/negative samples is a simple way to offset the unbalancing-problem, an additional experiment in Tab. 3 shows that our method performs better than assigning loss weight. The loss weights for positive/negative samples are determined by the positive/negative ratio in a mini-batch: $w_1 = \frac{\sum_i^{|Y|} 1(y_i==0)}{|Y|}$ and $w_0 = \frac{\sum_i^{|Y|} 1(y_i==1)}{|Y|}$, as suggested in [24].



(a) (b)

Figure 4: (a) Precision (dot), recall (dash dot), F-measure (solid line) and MaxF (circle) of DSS (red) and F-DSS (green) under different thresholds. Trained on unbalanced data, DSS tends to predict unknown pixels as the majority class–the background, resulting in high precision but low recall. FLoss is able to find a better compromise between precision and recall. (b) Precision, recall and F-measure of models trained by FLoss with different β^2. Precision decreases while recall increases with the rising of β^2 .

	Training data		ECSSD [25]			HKU-IS [10]			PASCALS [12]			SOD [17]			DUT-OMRON [17]		
Model	Train	#Images	MaxF	MeanF	MAE	MaxF	MeanF	MAE	MaxF	MeanF	MAE	MaxF	MeanF	MAE	MaxF	MeanF	MAE
DSS [7]	MB [14]	2.5K	0.908	0.889	0.060	0.899	0.877	0.048	0.824	0.806	0.099	0.835	0.815	0.125	0.761	0.738	0.071
DSS+Balance	MB [14]	2.5K	0.910	0.890	0.059	0.900	0.877	0.048	0.827	0.807	0.097	0.837	0.816	0.124	0.765	0.741	0.069
DSS+FLoss	MB [14]	2.5K	0.914	0.903	0.050	0.908	0.896	0.038	0.829	0.818	0.091	0.843	0.838	0.111	0.777	0.755	0.067

Table 3: Comparison of DSS, balanced-DSS and F-DSS.

4.5 The Compromise Between Precision and Recall

Recall and precision are two conflict metrics because high-precision predictions will usually achieve low recall and vice versa. In some circumstances, we care recall much more than precision, while in other tasks precision may be more important than recall. A representative example is the 'object proposal detection' which generates candidate bounding boxes for subsequent object classification. Recall must be firstly considered over precision in proposal generation because the missing candidates will no longer be retrieved.

We train models with different β^2 and comprehensively evaluate their performances in terms of precision, recall and F-measure. Results in Fig.4 (b) reveal that β^2 is a bias adjustor between precision and recall during model training: larger β^2 leads to higher recall while lower β^2 results in higher precision.

Conclusion

In this paper, we propose to directly maximize the F-measure for salient object detection. We introduce the FLoss that is differentiable w.r.t the predicted posteriors as optimization target in CNN architectures. The proposed method achieves better performance in terms of better handling the biased data distributions. Moreover, our method is stable against the threshold and obtains high-quality saliency maps under a wide threshold range, showing great potential in real-world applications. By adjusting the β^2 factor, one can easily adjust the compromise between precision and recall, adding more flexibility to deal with various circumstances. Comprehensive benchmarks on several popular datasets illustrate the advantage of the proposed method.

References

[1] Radhakrishna Achanta, Sheila Hemami, Francisco Estrada, and Sabine Susstrunk. Frequency-tuned salient region detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1597–1604. IEEE, 2009.

[2] Ali Borji, Ming-Ming Cheng, Qibin Hou, Huaizu Jiang, and Jia Li. Salient object detection: A survey. *Computer Vision and Image Understanding*, 2015.

[3] Ming-Ming Cheng, Niloy J Mitra, Xiaolei Huang, Philip HS Torr, and Shi-Min Hu. Global contrast based salient region detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(3):569–582, 2015.

[4] Krzysztof Dembczynski, Arkadiusz Jachnik, Wojciech Kotlowski, Willem Waegeman, and Eyke Hüllermeier. Optimizing the f-measure in multi-label classification: Plug-in rule approach versus structured loss minimization. In *International Conference on Machine Learning*, pages 1130–1138, 2013.

[5] Krzysztof J Dembczynski, Willem Waegeman, Weiwei Cheng, and Eyke Hüllermeier. An exact algorithm for f-measure maximization. In *Advances in neural information processing systems*, pages 1404–1412, 2011.

[6] Deng-Ping Fan, Jiangjiang Liu, Shanghua Gao, Qibin Hou, Ali Borji, and Ming-Ming Cheng. Salient objects in clutter: Bringing salient object detection to the foreground. *CoRR*, abs/1803.06091, 2018.

[7] Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip Torr. Deeply supervised salient object detection with short connections. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5300–5309. IEEE, 2017.

[8] Martin Jansche. A maximum expected utility framework for binary sequence labeling. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 736–743, 2007.

[9] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems*, pages 109–117, 2011.

[10] Guanbin Li and Yizhou Yu. Visual saliency based on multiscale deep features. *arXiv preprint arXiv:1503.08663*, 2015.

[11] Guanbin Li and Yizhou Yu. Deep contrast learning for salient object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 478–487, 2016.

[12] Yin Li, Xiaodi Hou, Christof Koch, James M Rehg, and Alan L Yuille. The secrets of salient object segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 280–287. Georgia Institute of Technology, 2014.

[13] Nian Liu and Junwei Han. Dhsnet: Deep hierarchical saliency network for salient object detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 678–686. IEEE, 2016.

[14] Tie Liu, Zejian Yuan, Jian Sun, Jingdong Wang, Nanning Zheng, Xiaoou Tang, and Heung-Yeung Shum. Learning to detect a salient object. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(2):353–367, 2011.

[15] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3431–3440, 2015.

[16] Ran Margolin, Lihi Zelnik-Manor, and Ayellet Tal. How to evaluate foreground maps? In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 248–255, 2014.

[17] Vida Movahedi and James H Elder. Design and perceptual validation of performance measures for salient object segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, pages 49–56. IEEE, 2010.

[18] Ye Nan, Kian Ming Chai, Wee Sun Lee, and Hai Leong Chieu. Optimizing f-measure: A tale of two approaches. *arXiv preprint arXiv:1206.4625*, 2012.

[19] James Petterson and Tibério S Caetano. Reverse multi-label learning. In *Advances in Neural Information Processing Systems*, pages 1912–1920, 2010.

[20] James Petterson and Tibério S Caetano. Submodular multi-label learning. In *Advances in Neural Information Processing Systems*, pages 1512–1520, 2011.

[21] José Ramón Quevedo, Oscar Luaces, and Antonio Bahamonde. Multilabel classifiers with a probabilistic thresholding strategy. *Pattern Recognition*, 45(2):876–883, 2012.

[22] Cornelis Joost Van Rijsbergen. Foundation of evaluation. *Journal of Documentation*, 30(4):365–373, 1974.

[23] Linzhao Wang, Lijun Wang, Huchuan Lu, Pingping Zhang, and Xiang Ruan. Saliency detection with recurrent fully convolutional networks. In *Eur. Conf. Comput. Vis.*, pages 825–841. Springer, 2016.

[24] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *Int. Conf. Comput. Vis.*, pages 1395–1403, 2015.

[25] Qiong Yan, Li Xu, Jianping Shi, and Jiaya Jia. Hierarchical saliency detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1155–1162. IEEE, 2013.

[26] Pingping Zhang, Dong Wang, Huchuan Lu, Hongyu Wang, and Xiang Ruan. Amulet: Aggregating multi-level convolutional features for salient object detection. *Int. Conf. Comput. Vis.*, 2017.