A Fully Convolutional Network for Salient Object Detection

Simone Bianco, Marco Buzzelli(✉), and Raimondo Schettini

Dipartimento di Informatica, Sistemistica e Comunicazione,
Università degli Studi di Milano-Bicocca, Viale Sarca 336, 20126 Milan, Italy
{simone.bianco,marco.buzzelli,schettini}@disco.unimib.it

Abstract. In this paper we address the task of salient object detection without requiring an explicit object class recognition. To this end, we propose a solution that exploits intermediate activations of a Fully Convolutional Neural Network previously trained for the recognition of 1,000 object classes, in order to gather generic object information at different levels of resolution. This is done by using both convolution and convolution-transpose layers, and combining their activations to generate a pixel-level salient object segmentation. Experiments are conducted on a standard benchmark that involves seven heterogeneous datasets. On average our solution outperforms the state of the art according to multiple evaluation measures.

Keywords: Salient object detection · Fully convolutional neural network · Foreground/background segmentation

1 Introduction

Accurate visual saliency models are fundamental for multiple disciplines such as computer vision [5], neuroscience [12], and cognitive psychology [11]. In this paper we focus on salient object detection, which consists in segmenting the main foreground object from the background in a digital image. Salient object detection methods are commonly used in applications such as object-of-interest proposal, object recognition, adaptive image and video compression, content aware image editing, image retrieval, and object-level image manipulation [6].

In the literature there is no universal agreement for the definition of foreground and background. This will be evident later on in this paper, by comparing the annotation criteria adopted through the different benchmark datasets. During the training of our Fully Convolutional Network, we simultaneously exploit annotated data coming from different datasets. In this way we obtain a model that represents a good compromise among the different datasets on the definition of foreground/background segmentation.

Many different approaches and solutions have been proposed in the last years for salient object detection. The method proposed in Discriminative Regional Feature Integration (DRFI) [13] builds a multi-level representation of the input

S. Battiato et al. (Eds.): ICIAP 2017, Part II, LNCS 10485, pp. 82–92, 2017.
https://doi.org/10.1007/978-3-319-68548-9_8

image, and creates a regression model mapping the regional feature vector of each level to the corresponding saliency score. These scores are finally fused in order to determine the complete saliency map. In Quantum Cut (QCUT) [3] authors model salient object segmentation as an optimization problem. They then exploit the link between quantum mechanics and graph-cuts to develop an object segmentation method based on the ground state solution of a modified Hamiltonian. The authors of Minimum Barrier Distance (MBD) [26] present an approximation of the MBD transform, and combine it with an appearance-based backgroundness cue. The resulting method performs significantly better than other solutions having the same computational requirements. In Saliency Tree (ST) [18] authors simplify the image into primitive regions, with associated saliency based on multiple handcrafted measures. They generate a saliency tree using region merging, and perform a systematic analysis of such tree to derive the final saliency map. Robust Background Detection (RBD) [28] introduces boundary connectivity: a background measure based on an intuitive geometrical interpretation. This measure is then used along with multiple low level cues to produce saliency maps through a principled optimization framework.

In a very recent work Borji et al. [6] present an exhaustive review of state of the art methods for salient object detection. They compared more than forty methods on a benchmark composed of seven different datasets. In this paper we investigate the use of a Fully Convolutional Network (FCN) for salient object detection taking inspiration from the work of Long et al. [19], and evaluate it on the Borji et al. [6] benchmark. Differently from compared solutions, we propose a data-driven model that leverages semantic cues as the basis for saliency estimation. Other approaches using deep learning methods also exist [7,10,15], although they don't adhere to the data and methods in the reference benchmark we adopt here. The main contributions of this paper can be summarized as follows:

- we propose a semantically-aware FCN to address the problem of salient object detection that is able to produce a binary pixel-level saliency map;
- we systematically investigate the contribution of different kinds of synthetic data augmentation to train the FCN;
- we evaluate the effectiveness of our proposal on a standard benchmark for salient object detection composed of seven different datasets [6]. The proposed method on average outperforms the state of the art according to multiple evaluation measures.

2 Proposed Method

We propose a Fully Convolutional Network to address the problem of salient object detection, taking inspiration from a work originally developed for semantic segmentation [19], that uses layers previously trained for the recognition of 1,000 object classes (Visual Geometry Group, or VGG [22]). This allows our network to be semantically-aware, and therefore capable of exploiting high-order concepts for separating foreground from background. Furthermore, the fully convolutional

architecture is specifically designed to produce a per-pixel prediction, which perfectly fits the task of generating an input-sized foreground/background mask.

The main difference with respect to the semantic segmentation proposed in [19] is that in our proposal the salient object could belong to any object category. Our network is in fact able to segment salient objects belonging to categories not restricted to the 20 classes defined in the original semantic segmentation task [19], or the 1,000 object classes used to train the VGG [22]. Finally, we adopt a different training procedure, as we find advantage in applying several kinds of data augmentation. The effects of such augmentation are analyzed and discussed in the experimental results section.

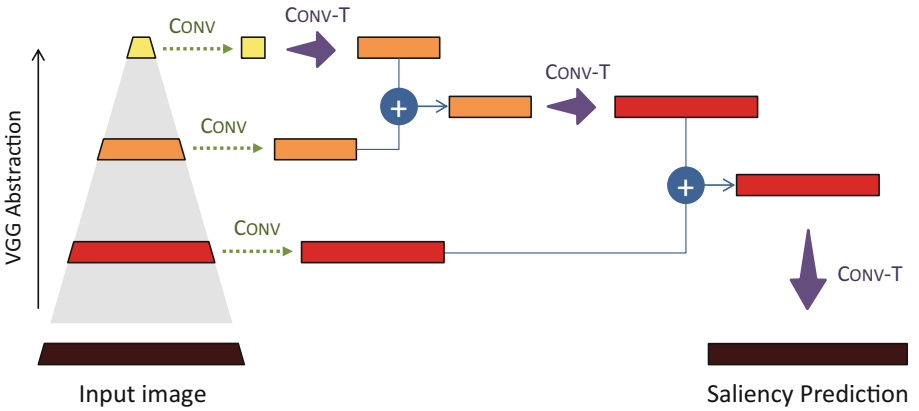


Fig. 1. Schematic view of the Fully Convolutional Network employed for salient object detection. Intermediate activations of a VGG-based processing are resized and combined in order to implement a multi-resolution analysis.

The network architecture is illustrated in Fig. 1, and adheres to the following logic:

1. Build abstractions of gradually decreasing spatial resolution, using [22].
2. Extract intermediate activations, and map their depth to the final problem size (2 classes for our task), using convolution layers.
3. Increase size of activations, using convolution-transpose layers.
4. Sum-up activations having now compatible size.
5. Produce as output a binary pixel-level saliency map.

Thanks to this strategy, the network can see both the whole picture and small details at the same time, thus producing a globally-aware yet precise output.

2.1 Training

Layers inherited from VGG (which supposedly only need fine-tuning) and new layers (trained from scratch), are all updated using the same learning-rate. The

task of calibrating the gradients for the two strategies is implicitly left to the Adam optimizer [14].

Many methods in the state of the art generate a continuous-valued prediction [3,13,18,26,28] directly correlated to the saliency of pixels in the image. Most of the available datasets, though, are published with a binary ground truth [5,8,17,24,25]. For this reason we choose to approach the problem as a per-pixel binary classification task: all ground truth images are converted to binary data, setting to 1 all values greater than 0. The neural network is then trained with a softmax cross entropy loss, with the global loss of each batch computed by averaging all loss values from the single pixels.

All training examples are processed by an online data augmentation procedure in order to provide additional information to the learning process. The following perturbations are considered:

- Random crop. We select a square subwindow of random side between 256 pixels and the original image limits. The crop is then resized to the fixed training dimension, i.e. 256×256 pixels.
- Random horizontal flip.
- Random gamma between 0.3 and $\frac{1}{0.3}$.

Each perturbation category was individually tested on a small subset of the benchmark data, in order to assess its impact on performance. An analysis on such effects is reported in Sect. 3.2.

All models are trained with a learning rate of 5×10^{-5} and a batch size equal to 15 for a total of 20 epochs.

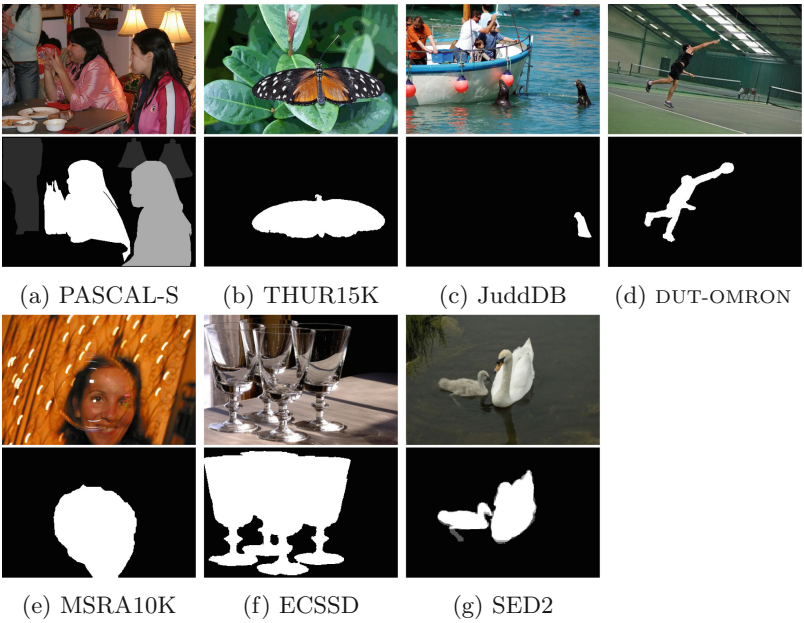
3 Experiments

3.1 Datasets

Experiments were performed according to the benchmark proposed in [6] concerning both the datasets and the evaluation protocol of the results. The benchmark is composed by seven different datasets that are presented in Table 1.

Table 1. Summary of tested datasets

Dataset	Images	Notes
PASCAL-S [16]	850	High background clutter
THUR15K [8]	6233	Only 6233/15000 annotated images
JuddDB [5]	900	Salient object typically very small
DUT-OMRON [25]	5166	-
MSRA10K [17]	10000	-
ECSSD [24]	1000	-
SED2 [2]	100	Two salient objects per picture



(a) PASCAL-S (b) THUR15K (c) JuddDB (d) DUT-OMRON

(e) MSRA10K (f) ECSSD (g) SED2

Fig. 2. Image-annotation examples for each of the seven datasets used in the benchmark [6].

Each dataset has different kinds of content and bias. Figure 2 shows an image-annotation pair for each dataset. The benchmark defines no official training/test split for the seven datasets, mainly because at the time of its original release few of the tested methods involved an explicit training phase. Our approach requires a significant amount of training data, so we adopted a Leave-One-Dataset-Out (LODO) solution. This allows us to have a fair comparison with the state of the art, as we test on the official datasets, and to avoid overfitting the model to the data. However, since in each LODO split we train the FCN on images collected and annotated with potentially different criteria than those used on the test set, our results could be lower than those we would obtain on homogeneous data (e.g. train/test split of the same dataset).

In order to ensure a totally fair evaluation procedure, we checked for any near-duplicates among dataset pairs. Following [4] we computed Structure Similarity measure (SSIM) [23] between all pairs of images, previously scaled to 64×64 pixels and converted to grayscale, and manually checked those having similarity higher than 0.9. Out of more than 200 million pairs, only five duplicates were found. Although this number of pairs is probably too small to have any overfitting effect, these images were excluded from the training set whenever the corresponding ones were present in the test set. Table 2 lists the found duplicate pairs.

Table 2. Duplicates found among the seven analyzed datasets.

Dataset/image		Dataset/Duplicate image
JuddDB/00854.jpg	≈	DUT-OMRON/sun_acnpbyuckesqygsf.jpg
PASCAL-S/101.jpg	≈	ECSSD/0046.jpg
PASCAL-S/180.jpg	≈	ECSSD/0054.jpg
PASCAL-S/276.jpg	≈	ECSSD/0062.jpg
PASCAL-S/277.jpg	≈	ECSSD/0063.jpg

3.2 Data Augmentation

A preliminary investigation on the usefulness of the data augmentation as described in Sect. 2.1 was performed on the DUT-OMRON Leave-One-Dataset-Out (LODO) setting. Figure 3 shows the loss values on both the training and test sets for three different setups: no data augmentation, three separate perturbations, and the same three perturbations applied jointly. It can be seen that all the investigated perturbation strategies reduce the ability of fitting the training data, while at the same time enhancing the model predictive power on unseen data. Their joint application results in the best improvement, thanks to the little correlation among the single contributions. Thus, it is used for the training of the FCN on all the datasets.

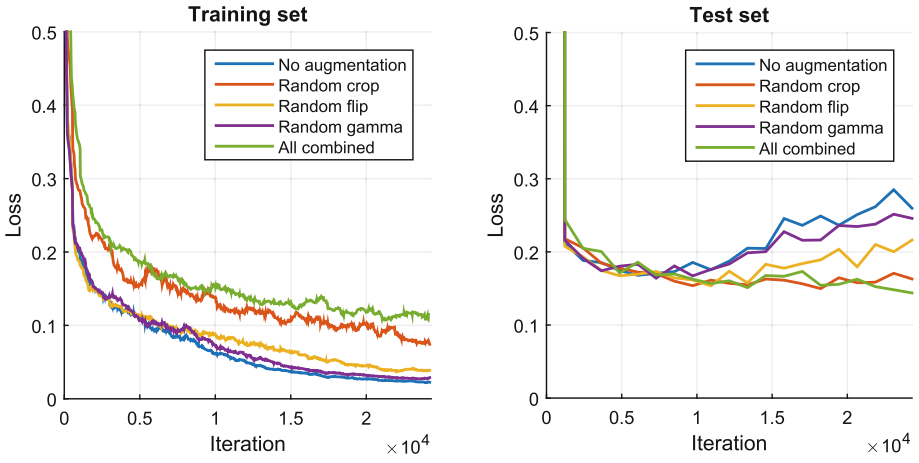


Fig. 3. Softmax cross entropy loss on the DUT-OMRON LODO setup under different kinds of data augmentation.

3.3 Evaluation Measures

Evaluation is performed under the following criteria, aimed at capturing different aspects of the quality of the predicted saliency region:

F-Measure (F_β) is the weighted harmonic mean between precision and recall:

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

According to [6] the weight β^2 is set to 0.3 in order to benefit precision, considered more important than recall for this specific task [1,17]. Since precision and recall require a binary input, the benchmark adopts three different alternatives for binarization of the methods that do not provide a binary prediction:

1. Varying fixed threshold: Precision and Recall are computed at all integer thresholds between 0 and 255, and then averaged.
2. Adaptive threshold [1]: The threshold for binarization is set to twice the mean value of the prediction map.
3. Saliency Cut [9]: The threshold is set to a low value, thus granting high recall rate. GrabCut [21] is then iteratively applied to the binarized prediction, typically producing a map with more precise edges.

Area Under Curve (AUC) is the area under the Receiver Operating Characteristic curve. The ROC curve is computed by varying the binarization threshold and plotting True Positive Rate (TPR) versus False Positive Rate (FPR) values.

Mean Absolute Error (MAE) is computed directly on the prediction, without any binarization step, as:

$$MAE = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} |Prediction(x,y) - GroundTruth(x,y)| \quad (2)$$

where W and H refer to image dimensions.

3.4 Results

We compare our solution with the top five methods from [6] on all the seven datasets using all criteria described in the previous section. Results are shown in Table 3.

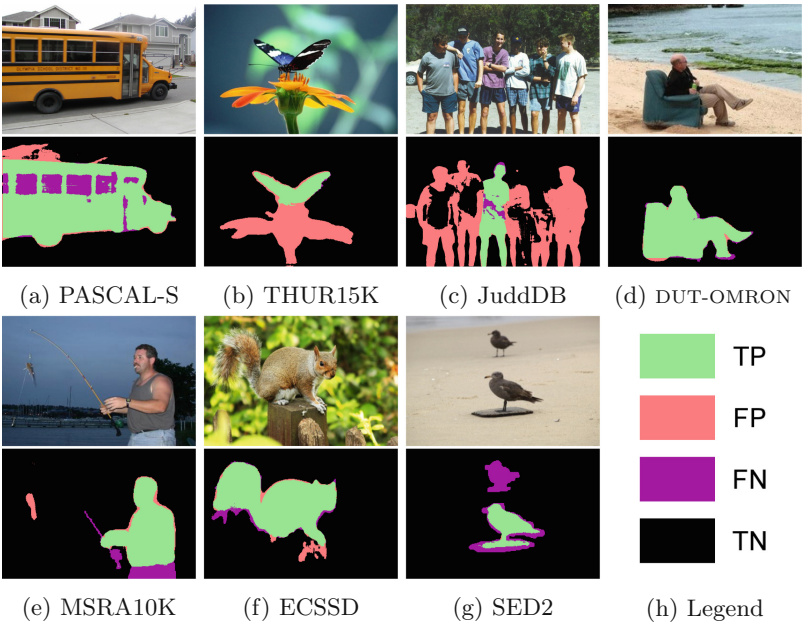
The proposed method is superior by a large margin according to both F_β measures and MAE. The binary nature of our prediction, though, is penalized by AUC due to the particular benchmark evaluation protocol [6]. On average our method outperforms all compared solutions for five of the seven datasets. On JuddDB and MSRA10K, and to a lesser extent on THUR15K, we have lower performance compared to the state of the art. We may notice that images in the JuddDB dataset contain many different subjects, out of which only one is annotated as the main salient object, based on fixations gathered from different observers. This particular set of conditions, radically different from those of the other datasets used for training in our Leave-One-Dataset-Out setup, could be the root cause of sub-optimal performance of our method, and it is left to future work for further analysis. Figure 4 reports some example predictions from all

Table 3. Evaluation results for all measures on all datasets

Measure	Method	P [16]	T [8]	J [5]	D [25]	S [2]	M [17]	E [24]	Average
F_β Varying	Ours	**0.763**	0.666	0.406	**0.706**	**0.847**	0.850	**0.864**	**0.729**
	DRFI [13]	0.679	**0.670**	0.475	0.665	0.831	**0.881**	0.787	0.713
	QCUT [3]	0.695	0.651	**0.509**	0.683	0.810	0.874	0.779	0.714
	MBD [26]	N/A	0.622	0.472	0.624	0.799	0.849	0.739	0.684
	ST [18]	0.660	0.631	0.455	0.631	0.818	0.868	0.752	0.688
	RBD [28]	0.652	0.596	0.457	0.630	0.837	0.856	0.718	0.678
Measure	Method	P [16]	T [8]	J [5]	D [25]	S [2]	M [17]	E [24]	Average
F_β Adaptive	Ours	**0.688**	0.620	0.382	**0.678**	**0.857**	0.833	**0.783**	**0.692**
	DRFI [13]	0.615	0.607	0.419	0.605	0.839	0.838	0.733	0.665
	QCUT [3]	0.654	**0.625**	**0.454**	0.647	0.801	**0.843**	0.738	0.680
	MBD [26]	N/A	0.594	0.422	0.592	0.803	0.830	0.703	0.657
	ST [18]	0.601	0.580	0.394	0.577	0.805	0.825	0.690	0.639
	RBD [28]	0.607	0.566	0.403	0.580	0.825	0.821	0.680	0.640
Measure	Method	P [16]	T [8]	J [5]	D [25]	S [2]	M [17]	E [24]	Average
F_β Sal Cut	Ours	**0.778**	**0.702**	0.409	**0.712**	**0.791**	0.890	**0.888**	**0.739**
	DRFI [13]	0.690	0.674	0.447	0.669	0.702	**0.905**	0.801	0.698
	QCUT [3]	0.613	0.620	**0.480**	0.647	0.672	0.843	0.747	0.660
	MBD [26]	N/A	0.642	0.470	0.636	0.759	0.890	0.785	0.697
	ST [18]	0.671	0.648	0.459	0.635	0.768	0.896	0.777	0.693
	RBD [28]	0.667	0.618	0.461	0.647	0.750	0.884	0.757	0.683
Measure	Method	P [16]	T [8]	J [5]	D [25]	S [2]	M [17]	E [24]	Average
AUC	Ours	0.820	0.851	0.680	0.828	0.844	0.877	0.896	0.828
	DRFI [13]	**0.897**	**0.938**	**0.851**	**0.933**	**0.944**	**0.978**	**0.944**	**0.926**
	QCUT [3]	0.870	0.907	0.831	0.897	0.860	0.956	0.909	0.890
	MBD [26]	N/A	0.915	0.838	0.903	0.922	0.964	0.917	0.910
	ST [18]	0.868	0.911	0.806	0.895	0.922	0.961	0.914	0.897
	RBD [28]	0.867	0.887	0.826	0.894	0.899	0.955	0.894	0.889
Measure	Method	P [16]	T [8]	J [5]	D [25]	S [2]	M [17]	E [24]	Average
MAE	Ours	**0.122**	**0.106**	**0.210**	**0.079**	**0.080**	**0.073**	**0.065**	**0.105**
	DRFI [13]	0.221	0.150	0.213	0.155	0.130	0.118	0.166	0.165
	QCUT [3]	0.195	0.128	0.178	0.119	0.148	0.118	0.171	0.151
	MBD [26]	N/A	0.162	0.225	0.168	0.137	0.107	0.172	0.162
	ST [18]	0.224	0.179	0.240	0.182	0.145	0.122	0.193	0.184
	RBD [28]	0.199	0.150	0.212	0.144	0.130	0.108	0.173	0.159

datasets. False positives mostly correspond to actual objects that were not in the ground truth due to annotation guidelines (e.g. the flower in Fig. 4b and the fish in Fig. 4e), which could also be contributing to the lower performance on datasets MSRA10K and THUR15K. False negatives are often related to holes in our prediction (e.g. the window glasses in Fig. 4a), thus highlighting a current

downside of the solution. Finally, we can also observe that the edges of our predictions are in general smoother and less precise than the reference annotations.



(a) PASCAL-S (b) THUR15K (c) JuddDB (d) DUT-OMRON

(e) MSRA10K (f) ECSSD (g) SED2 (h) Legend

Fig. 4. Example predictions on different datasets.

A direct comparison with other methods in terms of computational complexity cannot be performed in a fair setup, as our solution is designed to run on GPU, unlike the compared methods. On a NVIDIA TITAN X GPU our prediction takes on average 0.09 s on each image of the MSRA10K dataset (typical image resolution 400×300). For reference, the fastest among compared methods (RBD [28]) takes 0.269 s using a desktop machine with Xeon E5645 2.4 GHz CPU [6].

4 Conclusions

In this work we exploited the semantic awareness of a Fully Convolutional Network to address the problem of salient object detection. We verified the effectiveness of this approach by comparing it on a standard benchmark, composed of seven datasets and more than forty methods (we reported here only the top five). Despite the challenging Leave-One-Dataset-Out setup, which naturally excludes the possibility of overfitting the model to the data, we outperformed the state of the art on most datasets.

In the future we might switch from a binary foreground/background prediction to a multiclass one, in order to also consider the different levels of saliency

defined in some of the used datasets. Bringing this even further, we might directly treat the problem as a regression task, and study the effects of different training losses on the final performance.

Finally, we plan on extending evaluation and comparison to other datasets [15,20] and methods [15,27], which were currently left-out for not being contemplated in the adopted benchmark, as well as for space constraints.

References

1. Achanta, R., Hemami, S., Estrada, F., Susstrunk, S.: Frequency-tuned salient region detection. In: Computer vision and pattern recognition. In: IEEE Conference on VPR 2009, pp. 1597–1604. IEEE (2009)
2. Alpert, S., Galun, M., Brandt, A., Basri, R.: Image segmentation by probabilistic bottom-up aggregation and cue integration. IEEE Trans. Pattern Anal. Mach. Intell. **34**(2), 315–327 (2012)
3. Aytekin, C., Kiranyaz, S., Gabbouj, M.: Automatic object segmentation by quantum cuts. In: 2014 22nd International Conference on Pattern Recognition (ICPR), pp. 112–117. IEEE (2014)
4. Bianco, S., Buzzelli, M., Mazzini, D., Schettini, R.: Deep learning for logo recognition. Neurocomputing **245**, 23–30 (2017). http://dx.doi.org/10.1016/j.neucom.2017.03.051
5. Borji, A.: What is a salient object? a dataset and a baseline model for salient object detection. IEEE Trans. Image Process. **24**(2), 742–756 (2015)
6. Borji, A., Cheng, M.M., Jiang, H., Li, J.: Salient object detection: a benchmark. IEEE Trans. Image Process. **24**(12), 5706–5722 (2015)
7. Chen, T., Lin, L., Liu, L., Luo, X., Li, X.: Disc: deep image saliency computing via progressive representation learning. IEEE Trans. Neural Netw. Learn. Syst. **27**(6), 1135–1149 (2016)
8. Cheng, M.M., Mitra, N.J., Huang, X., Hu, S.M.: Salientshape: group saliency in image collections. Vis. Comput. **30**(4), 443–453 (2014)
9. Cheng, M.M., Mitra, N.J., Huang, X., Torr, P.H., Hu, S.M.: Global contrast based salient region detection. IEEE Trans. Pattern Anal. Mach. Intell. **37**(3), 569–582 (2015)
10. Han, J., Zhang, D., Hu, X., Guo, L., Ren, J., Wu, F.: Background prior-based salient object detection via deep reconstruction residual. IEEE Trans. Circ. Syst. Video Technol. **25**(8), 1309–1321 (2015)
11. Hayhoe, M., Ballard, D.: Eye movements in natural behavior. Trends Cogn. Sci. **9**(4), 188–194 (2005)
12. Itti, L., Koch, C.: Computational modelling of visual attention. Nat. Rev. Neurosci. **2**(3), 194–203 (2001)
13. Jiang, H., Wang, J., Yuan, Z., Wu, Y., Zheng, N., Li, S.: Salient object detection: a discriminative regional feature integration approach. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2083–2090 (2013)
14. Kingma, D., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint (2014). arXiv:1412.6980
15. Li, G., Yu, Y.: Visual saliency based on multiscale deep features. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5455–5463 (2015)

16. Li, Y., Hou, X., Koch, C., Rehg, J.M., Yuille, A.L.: The secrets of salient object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 280–287 (2014)
17. Liu, T., Yuan, Z., Sun, J., Wang, J., Zheng, N., Tang, X., Shum, H.Y.: Learning to detect a salient object. IEEE Trans. Pattern Anal. Mach. Intell. **33**(2), 353–367 (2011)
18. Liu, Z., Zou, W., Le Meur, O.: Saliency tree: a novel saliency detection framework. IEEE Trans. Image Process. **23**(5), 1937–1952 (2014)
19. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
20. Movahedi, V., Elder, J.H.: Design and perceptual validation of performance measures for salient object segmentation. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 49–56. IEEE (2010)
21. Rother, C., Kolmogorov, V., Blake, A.: Grabcut: interactive foreground extraction using iterated graph cuts. In: ACM Transactions on Graphics (TOG), vol. 23, pp. 309–314. ACM (2004)
22. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint (2014). arXiv:1409.1556
23. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)
24. Yan, Q., Xu, L., Shi, J., Jia, J.: Hierarchical saliency detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1155–1162 (2013)
25. Yang, C., Zhang, L., Lu, H., Ruan, X., Yang, M.H.: Saliency detection via graph-based manifold ranking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3166–3173 (2013)
26. Zhang, J., Sclaroff, S., Lin, Z., Shen, X., Price, B., Mech, R.: Minimum barrier salient object detection at 80 fps. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1404–1412 (2015)
27. Zhao, R., Ouyang, W., Li, H., Wang, X.: Saliency detection by multi-context deep learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1265–1274 (2015)
28. Zhu, W., Liang, S., Wei, Y., Sun, J.: Saliency optimization from robust background detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2814–2821 (2014)