



Combining Saliency Estimation Methods

Marco Buzzelli^(✉), Simone Bianco, and Gianluigi Ciocca

University of Milano - Bicocca, Milan, Italy
{marco.buzzelli,simone.bianco,gianluigi.ciocca}@unimib.it

Abstract. We address the task of image saliency estimation through proper recombination of existing methods in the state of the art. We define a general scheme, which we then specialize to perform dataset-specific and image-specific recombination, based on either linear weight regression, or method selection. The advantage of this approach lies in the possibility of exploiting the different strengths of existing methods. Experiments are conducted with both deep learning and hand-crafted methods on a widely used dataset, using standard evaluation measures. The proposed recombination strategy allows us to improve upon the state of the art, by exploiting a linear combination of the saliency maps produced by existing methods. We also show that image-specific combination and selection of saliency maps is limited by the apparent lack of relevant information intrinsic in the image itself.

Keywords: Saliency estimation · Combining · Deep learning

1 Introduction

Saliency estimation refers to the localization of the areas in an image having particular clue for a human observer, while salient object detection refers to the detection and segmentation of the most salient objects in the scene. There is, however, no consensus about the definition of “what saliency is” in the community. Multiple observers may consider salient different elements in the scene, and some elements may be considered more salient than others depending on the scene context and/or on the observer’s cultural background. This makes saliency estimation an ill-posed problem [1, 29]. This is also reflected by the many saliency detection methods proposed in the literature. As demonstrated by the authors in [6] and [10], there is no best overall saliency detection algorithm that is able to achieve equally good results across different benchmark datasets. They analyze and benchmark many different saliency detection algorithms each based on different assumptions, heuristics and features that can be either hand-crafted, learned by Convolutional Neural Networks (CNN), or both.

Among the hand-crafted approaches is [17] that computes saliency from the perspective of image reconstruction error of background images generated at different level of details. A graph-based manifold ranking is used in [30] to classify superpixels into foreground and background regions. In [34] an image patch is

considered not salient if it is heavily connected to the image boundaries. Other graph-based approaches are the ones presented in [11], and [2]. In [32] is presented a graph-based approach that exploits a fast Minimum Barrier Distance to measure a pixel’s connectivity to the image boundary. In [7] the saliency of each image region is carried out by simultaneously evaluating global contrast differences and spatial coherence with nearby regions. Color is used in [12] where the saliency is based on a linear combination of high-dimensional color spaces. In [22] global contrast, spatial sparsity, and object priors are integrated to estimate the saliency of image regions. Finally, an approach based on multiple features computed in a multi-level segmentation schema is presented in [26].

CNN-based approaches are able to process images extracting information at different levels of details, and can automatically learn what is the relevant information within an image given a specific task. For example, a multi-branch approach is proposed in [33] and [15], processing the image at a different level of details. In [16] the image is analyzed to produce pixel-level and super-pixel level segmentation maps that are then fused together, while a multi-task learning scheme based on saliency and segmentation is used in [18]. In [19] it is designed a novel network architecture that works in a global-to-local manner to improve saliency detection performance. In [14] both low level and high level features are exploited in a unified deep learning approach. A CNN is used in [31] as an embedding function to map pixels and their attributes to classify them as salient/background. Using different layers in the neural architecture provides multi-scale feature maps that can be exploited for an efficient salient object detection. Example of algorithms using this approach are [10], and [3]. Recurrent network architectures can effectively help reducing prediction errors by iteratively integrating contextual information which is important for saliency detection. To this end, recurrent convolutional networks are used to refine the saliency map by correcting errors during the learning process [20,27,28].

2 Proposed Approach to Saliency Estimation

Our approach to saliency estimation is based on the analysis, selection, and combination of existing saliency estimation methods.

Given an input image $i \in I$, and a set $S_i = \{s_{i,m} : m \in M\}$ of saliency predictions produced by $|M|$ existing saliency estimation methods, we construct a novel saliency map \hat{s}_i by linear combination:

$$\hat{s}_i = \left(\sum_{m \in M} w_{i,m} \cdot s_{i,m} \right) \geq T \quad (1)$$

where $w_{i,m} \in \mathbb{R}$, and $T=0.5$. Imposing a threshold T on the predictions eventually produces a binary estimation of image saliency, which has been shown to positively affect standard measures [3].

We introduce various additional constraints into Eq. 1, that allow us to frame the problem in terms of either combination of saliency methods, or selection of saliency methods.

2.1 Combination of Saliency Methods

Equation 1 is formulated in terms of linear combination of weights. A Convolutional Neural Network (CNN) is trained to generate the proper set of weights by processing either the RGB image itself, or the related existing saliency estimations. To this extent, preliminary investigation led to the adoption of a ResNet-18 [9], trained with the objective of reproducing the ground truth saliency maps. The hard threshold T cannot be directly applied during the training process, as its non-differentiability would compromise the gradient backpropagation. It has been instead replaced, at training-time only, with a soft threshold implemented through a steep sigmoid function. On top of this solution, we experiment with further constraints:

1. We impose a fixed set of weights for all images:

$$w_{i,m} = w_{j,m}, \quad \forall i, j \in I \quad (2)$$

In this case, the weights will be optimized globally on the defined training set, instead of learning to infer them from each RGB image.

2. We limit the linear combination to $M^{(N)}$, defined as the subset of the N best-performing methods. That is:

$$w_{i,m} = 0, \quad \forall m \in M \setminus M^{(N)} \quad (3)$$

2.2 Selection of Saliency Methods

The weight-regression problem defined in the previous sections can be reformulated as a classification task:

$$\sum_{m \in M} w_{i,m} = 1, \quad w_{i,m} \in \{0, 1\} \quad (4)$$

To this extent, a CNN is trained with the objective of selecting, for each image, the best performing method. In this case, the model is trained with a softmax cross-entropy loss, comparing the performed selection with the defined best method. The ground truth best method can be determined as the one with the best performance evaluated with standard measures (see Sect. 3.1). The input to the neural model can either be the RGB image itself, or the related existing saliency maps.

Existing literature covers this classification-oriented view of the problem as an Ensemble Dictionary Learning task (EDL) [35]. Here we focus on whole-image selection of more-recently developed methods for saliency estimation.

3 Experimental Results

In this section we present the experimental setup, along with results obtained on saliency estimation with the proposed combination and selection of existing solutions.

3.1 Experimental Setup

State of the art methods for saliency estimation are often published along with the corresponding predictions on standard datasets. The most popular ones are the MSRAB dataset [21] and the related MSRA10K dataset [7]. In order to setup a common dataset among all algorithms exploited in this paper, we defined the MSRAB Validation Subset (366 images) and MSRAB Test Subset (1516 images), as the intersection between the original splits of MSRAB and the entire MSRA10K dataset. For the optimization of a fixed set of weights, we train on the MSRAB Validation Subset and evaluate on the MSRAB Test Subset. To learn the more complex image-specific models, instead, we train on a combination of existing datasets for saliency estimation from [3], and also evaluate on the MSRAB Test Subset. Following [10], evaluation is performed in terms of both MAE and F_β :

$$MAE = \frac{1}{|I|} \sum_{i \in I} \frac{1}{|C|} \sum_{c \in C} |PR_{i,c} - GT_{i,c}| \quad (5)$$

$$F_{\beta=\sqrt{0.3}} = \max_{t \in T} \frac{(1 + \beta^2) \frac{1}{|I|} \sum_{i \in I} Precision_i(t) \cdot \frac{1}{|I|} \sum_{i \in I} Recall_i(t)}{\beta^2 \cdot \frac{1}{|I|} \sum_{i \in I} Precision_i(t) + \frac{1}{|I|} \sum_{i \in I} Recall_i(t)} \quad (6)$$

where I is the set of images, C the set of coordinates for every given image, and T the set of possible thresholds. PR and GT are, respectively, the saliency prediction and ground truth. Expansion of *Precision* and *Recall* is here omitted for brevity reasons. A mixture measure can also be defined as the average of the complemented rescaled MAE , and the rescaled F_β :

$$mix_{i,m} = (1 - norm_{i,m}^{MAE}) + norm_{i,m}^{F_\beta} \quad (7)$$

where:

$$norm_{i,m}^x = \frac{x_{i,m} - \min_{m \in M}(x_{i,m})}{\max_{m \in M}(x_{i,m}) - \min_{m \in M}(x_{i,m})} \quad (8)$$

In this work we considered a total of 20 saliency estimation algorithms: ten hand-crafted, and ten deep-based. For the hand-crafted ones we analyzed SC [33], DHS [19], MDF [15], ELD [14], DS [18], DCL [16], RFCN [27], DRCN [20], DSS [10], and MFCN [3]. For the had-crafted algorithms, we considered GMR [30], DSR [17], MC [11], ST [22], RBD [34], EQC [2], MB+ [32], RC [7], HDCT [12], and RFI [26].

3.2 Dataset Content Analysis

Before evaluating the combination strategies, we investigated if there is any connection between image content and best performing algorithm. The first column in Table 1 shows a priori probabilities of each method (belonging to either the deep learning, or hand-crafted family) being the best one for the analyzed

Table 1. A priori and conditioned probabilities of different methods (M) given a certain type of content (belonging to either C_1 and C_2). Algorithms are sorted in descending order by the mixture measure defined in Eq. 7.

	Method	$P(M)$	$P(M C_1)$			$P(M C_2)$	
			Object	Scene/in	Scene/out	No-people	People
Deep learning (DL)	DHS [19]	0.553	0.517	0.709	0.593	0.553	0.553
	ELD [14]	0.193	0.249	0.109	0.117	0.206	0.142
	DSS [10]	0.144	0.118	0.073	0.190	0.130	0.198
	RFCN [27]	0.012	0.018	0.000	0.003	0.013	0.006
	DCL [16]	0.046	0.038	0.036	0.059	0.044	0.053
	DS [18]	0.003	0.002	0.018	0.002	0.003	0.003
	MFCN [3]	0.007	0.008	0.000	0.005	0.006	0.009
	SC [33]	0.001	0.002	0.000	0.000	0.002	0.000
	MDF [15]	0.033	0.041	0.055	0.019	0.037	0.019
	DRCN [20]	0.008	0.006	0.000	0.012	0.006	0.016
Hand-crafted (HC)	EQC [2]	0.201	0.193	0.200	0.214	0.216	0.145
	DRFI [26]	0.119	0.104	0.127	0.139	0.114	0.138
	ST [22]	0.171	0.159	0.200	0.185	0.169	0.179
	MB+ [32]	0.108	0.097	0.091	0.126	0.105	0.119
	RBD [34]	0.113	0.139	0.055	0.080	0.119	0.088
	GMR [30]	0.123	0.148	0.182	0.081	0.125	0.116
	DSR [17]	0.068	0.057	0.073	0.083	0.063	0.085
	RC [7]	0.030	0.026	0.000	0.039	0.028	0.041
	MC [11]	0.035	0.052	0.018	0.012	0.040	0.016
	HDCT [12]	0.032	0.024	0.055	0.041	0.021	0.072

dataset. We can see how these distributions change when conditioned on two types of image content: the first conditioning (C_1) partitions the possible subjects into “object”, “scene (indoor)”, and “scene (outdoor)”. The second conditioning (C_2) considers the presence or absence of people in the image. We observe little impact on the probability distribution over M with or without different types of conditioning. This suggests little to no connection between the considered image content and best performing method.

3.3 Combination of Saliency Methods

The first experiment consists in determining a set of linear combination weights specific for each input image. This has been dealt with by defining a CNN that predicts the linear weights as a function of either the RGB image itself, or the saliency estimation maps produced by existing saliency estimation methods.

The second experiment consists in evaluating a dataset-specific weights combination instead of image-specific ones. In order to do this, we optimize a fixed set of weights in Eq. 1 to be applied for all the images. As shown in the third and fourth rows of Table 2, for this experiment the best results were obtained on deep learning methods using as input the RGB images, producing 0.0253 MAE and 0.9418 F_β . These values represent an improvement with respect to the reported baselines in the first and second rows of Table 2: best single method and combination with uniform weights. The best single method refers, respectively, to

Table 2. Performance obtained with method combination on the MSRAB Test Subset.

	Deep learning (DL)		Hand-crafted (HC)	
	MAE	F_β	MAE	F_β
Single best method (DHS [19] and EQC [2])	0.0275	0.9365	0.0927	0.8365
Uniform weights	0.0265	0.9407	0.0715	0.8803
Image-specific weights (from RGB)	0.0253	0.9418	0.0601	0.8928
Image-specific weights (from saliency)	0.0254	0.9426	0.0590	0.8953
Dataset-specific weights (fixed)	0.0242	0.9445	0.0528	0.8887

DHS for deep learning solutions and EQC for what concerns hand-crafted algorithms. The fixed weights optimization allowed reaching 0.242 MAE and 0.9445 F_β on deep learning methods, compared to the best single-method DHS (0.0275 MAE and 0.9365 F_β). Significant improvements are also observed on hand-crafted methods, although the obtained performance does not reach the level of deep learning solutions. As a general observation, the simpler fixed-weights setup appears to outperform the generation of image-specific weights. We hypothesize that the RGB images do not contain enough information to provide the necessary nuanced image-specific sets of weights.

To further explore the promising fixed-weights setup, we optimize on a varying subset of saliency estimation methods. Figure 1 shows the performance obtained, in terms of MAE and F_β , by imposing zero weights on all but the first N best performing methods, as ordered as in Table 1. As a comparison, uniform weights are also reported. It can be seen that introducing more than four best-performing methods in a uniform-weight linear combination, deteriorates the overall performance. This is especially evident for what concerns the deep learning methods (sub-figures (a) and (b)). Corroborating this observation, the improvement introduced by proper weight optimization on less performing methods appears to be negligible. Notice that the curve trend on the training set is not strictly decreasing due to the optimization being guided by a mixture of the two metrics, and due to the randomness of the mini-batch training process.

3.4 Selection of Saliency Methods

In this Section, we present an analysis on image-specific selection of saliency estimation methods. To this extent, Table 3 offers several baselines: uniform sampling is a purely random selection of the input saliency estimation method, while prior sampling takes into account the *a priori* probability of each method being the best solution on the MSRAB Test Subset. It should be noted that the prior sampling under-performs with respect to the best single method, due to the adopted metrics not being directly related to classification accuracy. An ideal oracle could be based on MAE , F_β , or a mix of the two measures, as shown in rows four to six in Table 3, reaching in the best scenario 0.0200 MAE and 0.9553 F_β for deep learning methods. These values can therefore be considered the upper-bound of any solution for automated selection of image-specific saliency estimation methods.

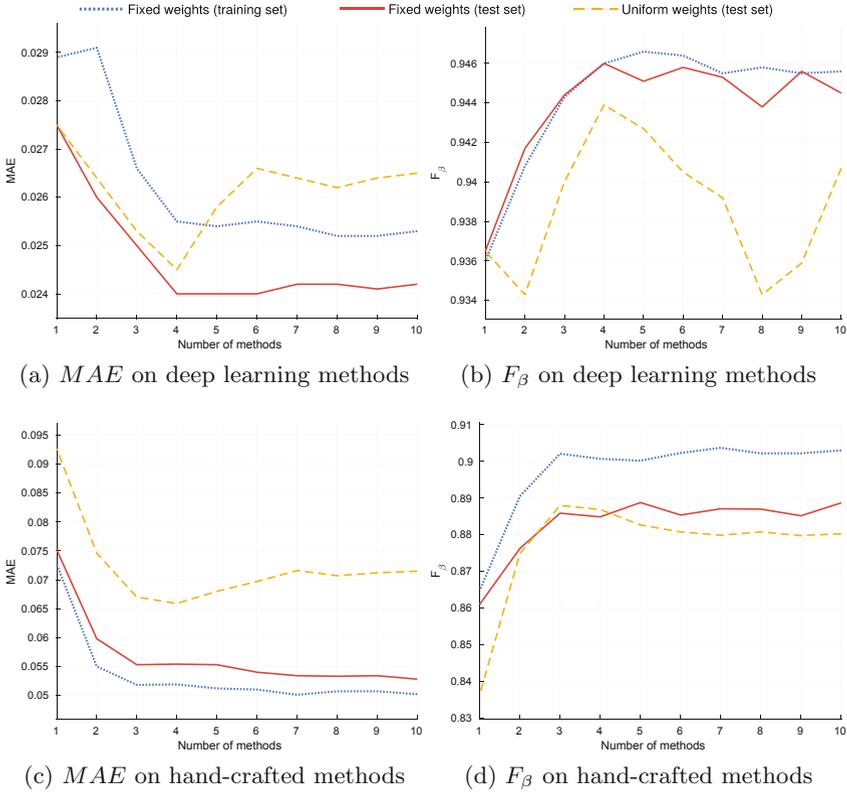


Fig. 1. Performance of fixed weights on the MSRAB Validation Subset and Test Subset, obtained by optimizing on only the first N best performing saliency estimation methods.

For this experiment, a CNN has been trained with the task of replicating the effect of each mix-based oracle, in selecting the best saliency estimation method for each specific image, based on a processing of the RGB image or the existing saliency estimation maps. The results reported in the last two rows of Table 3 show that, for deep learning solutions, only a small improvement over the best single method can be obtained by analyzing the available saliency maps, while the RGB images do not contain enough information to perform the task. This is in line with what has already been observed for methods combination. In the case of hand-crafted methods, the results obtained by the trained neural models outperform all reported baselines (Fig. 2).

3.5 Embedding Analysis

In this section we want to assess if indeed the RGB images contain enough information to capture the necessary nuanced for the optimization of image-specific weights (Sect. 3.3) or to predict the best method to apply (Sect. 3.4). To

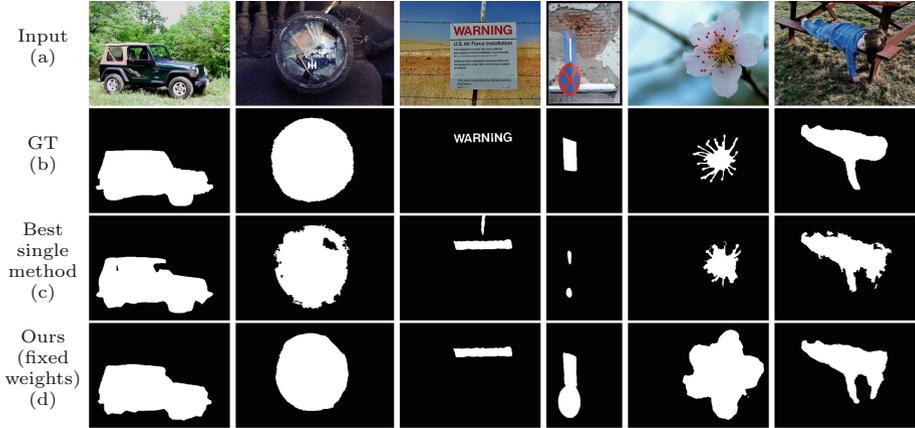


Fig. 2. Visual results of our approach to combination of saliency estimation maps. Rows (a) and (b) show, respectively, the starting image and the corresponding ground truth annotation. Row (c) is the best single method solution. Row (d) is our fixed-weight optimization.

Table 3. Performance obtained with method selection on the MSRAB Test Subset.

	Deep learning (DL)		Hand-crafted (HC)	
	MAE	F_{β}	MAE	F_{β}
Single best method (DHS [19] and EQC [2])	0.0275	0.9365	0.0927	0.8365
Uniform sampling	0.0734	0.8632	0.0889	0.8406
Prior sampling	0.0355	0.9213	0.0863	0.8444
Oracle (MAE)	0.0173	0.9538	0.0473	0.9105
Oracle (F_{β})	0.0200	0.9521	0.0610	0.8997
Oracle (mix)	0.0175	0.9553	0.0465	0.9171
Image-specific selection (from RGB)	0.0298	0.9319	0.0795	0.8565
Image-specific selection (from saliency)	0.0270	0.9368	0.0835	0.8524

this end, we extract a set of features that describe images from different points

Table 4. 1-NN classification performance of the different features considered.

Features	Accuracy (macro)	Accuracy (micro)
Single best method (DHS [19])	0.4677	0.1000
RGB	0.2863	0.1066
RGB histogram	0.2850	0.1060
LBP	0.2876	0.1206
HOG	0.2559	0.0945
AlexNet	0.3008	0.1209
Inception-ResNet-v2	0.3047	0.1200

of view: simple RGB statistics (channel average and standard deviation), RGB histograms (concatenation of the channel histograms each with 16 bins), LBP [24], HOG [8], CNN features from two models trained on ImageNet: AlexNet [13], chosen for its popularity, and Inception-ResNet-v2 [25], chosen for its good balance between accuracy and number of operations [4]. Further experiments might also take into account local descriptors [5]. The analysis is performed by creating 2D projections of each feature with t-SNE [23]. The projections are reported in Fig. 3. Each point represents the projection of a feature extracted from one image, and its color corresponds to the best method for that given image. From the projections it is possible to notice that for certain features some clusters emerge. In the ideal case, an informative feature, would create a separate cluster for each method. In order to measure the purity of the clusters created, we perform a 1-NN classification on each feature. The classification results in terms of both macro-averaged and micro-averaged accuracy are reported in Table 4. As a further comparison, we also add the performance of the classifiers that always predicts to use the global best method (i.e. DHS). From the results reported it is possible to notice that the best results are obtained by the CNN features, but that the results are very low, suggesting a large impurity in the clusters thus enforcing the hypothesis that the RGB images do not contain enough information (or do not provide suitable information) to perform this task.

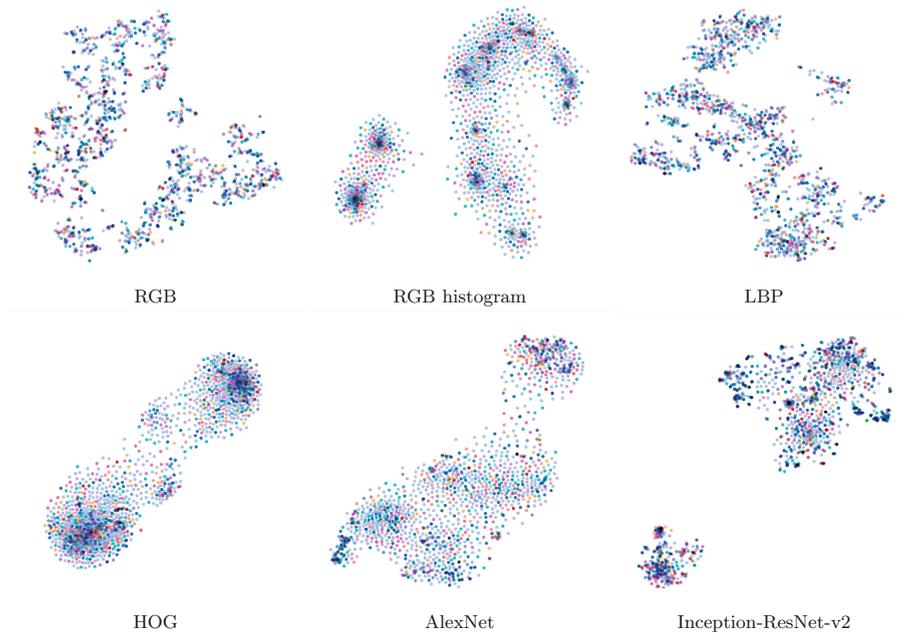


Fig. 3. 2D t-SNE [23] projections of each feature considered. (Color figure online)

4 Conclusions

By considering all the different rationales in existing saliency estimation algorithms, and their diverse levels of performance, we define a general scheme for saliency estimation through proper recombination of existing methods in the state of the art. The advantage of this approach lies in the possibility of exploiting the different strengths of existing methods. We treat the combination problem as either linear weight regression, or method selection. We are able to improve performance on the state of the art by optimizing a linear combination over a subset of saliency estimation method. Several attempts at producing image-specific combination resulted in sub-optimal results. Further analysis showed the apparent non-correlation between image content and best performing saliency estimation algorithm for both the combination and selection tasks.

Acknowledgments. The research leading to these results has received funding from TEINVEIN: CUP: E96D17000110009, cofunded by POR FESR 2014–2020.

References

1. Amirul Islam, M., Kalash, M., Bruce, N.D.: Revisiting salient object detection: simultaneous detection, ranking, and subitizing of multiple salient objects. In: IEEE CVPR, pp. 7142–7150 (2018)
2. Aytekin, Ç., Ozan, E.C., Kiranyaz, S., Gabbouj, M.: Visual saliency by extended quantum cuts. In: IEEE ICIP, pp. 1692–1696 (2015)
3. Bianco, S., Buzzelli, M., Schettini, R.: Multiscale fully convolutional network for image saliency. *JEI* **27**, 051221 (2018)
4. Bianco, S., Cadene, R., Celona, L., Napoletano, P.: Benchmark analysis of representative deep neural network architectures. *IEEE Access* **6**, 64270–64277 (2018)
5. Bianco, S., Schettini, R., Mazzini, D., Pau, D.P.: Quantitative review of local descriptors for visual search. In: IEEE Third ICCE-Berlin, pp. 98–102. IEEE (2013)
6. Borji, A., Cheng, M., Jiang, H., Li, J.: Salient object detection: a benchmark. *IEEE TIP* **24**(12), 5706–5722 (2015)
7. Cheng, M., Mitra, N.J., Huang, X., Torr, P.H.S., Hu, S.: Global contrast based salient region detection. *IEEE TPAMI* **37**(3), 569–582 (2015)
8. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE CVPR, vol. 1, pp. 886–893. IEEE Computer Society (2005)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE CVPR, pp. 770–778 (2016)
10. Hou, Q., Cheng, M., Hu, X., Borji, A., Tu, Z., Torr, P.H.S.: Deeply supervised salient object detection with short connections. *IEEE TPAMI* **41**, 815–828 (2018)
11. Jiang, B., Zhang, L., Lu, H., Yang, C., Yang, M.H.: Saliency detection via absorbing markov chain. In: IEEE ICCV, pp. 1665–1672 (2013)
12. Kim, J., Han, D., Tai, Y., Kim, J.: Salient region detection via high-dimensional color transform and local spatial support. *IEEE TIP* **25**(1), 9–23 (2016)
13. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS, pp. 1097–1105 (2012)
14. Lee, G., Tai, Y.W., Kim, J.: Deep saliency with encoded low level distance map and high level features. In: IEEE CVPR, pp. 660–668 (2016)

15. Li, G., Yu, Y.: Visual saliency based on multiscale deep features. In: IEEE CVPR, pp. 5455–5463 (2015)
16. Li, G., Yu, Y.: Deep contrast learning for salient object detection. In: IEEE CVPR, pp. 478–487 (2016)
17. Li, X., Lu, H., Zhang, L., Ruan, X., Yang, M.: Saliency detection via dense and sparse reconstruction. In: ICCV, pp. 2976–2983 (2013)
18. Li, X., et al.: Deepsaliency: multi-task deep neural network model for salient object detection. IEEE TIP **25**(8), 3919–3930 (2016)
19. Liu, N., Han, J.: DHSNet: deep hierarchical saliency network for salient object detection. In: IEEE CVPR, pp. 678–686 (2016)
20. Liu, N., Han, J.: A deep spatial contextual long-term recurrent convolutional network for saliency detection. IEEE TIP **27**(7), 3264–3274 (2018)
21. Liu, T., et al.: Learning to detect a salient object. IEEE TPAMI **33**(2), 353–367 (2011)
22. Liu, Z., Zou, W., Meur, O.L.: Saliency tree: a novel saliency detection framework. IEEE TIP **23**(5), 1937–1952 (2014)
23. van der Maaten, L., Hinton, G.: Visualizing data using t-SNE. J. Mach. Learn. Res. **9**(1), 2579–2605 (2008)
24. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE TPAMI **7**, 971–987 (2002)
25. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: AAAI Conference on AI (2017)
26. Wang, J., Jiang, H., Yuan, Z., Cheng, M.M., Hu, X., Zheng, N.: Salient object detection: a discriminative regional feature integration approach. IJCV **123**(2), 251–268 (2017)
27. Wang, L., Wang, L., Lu, H., Zhang, P., Ruan, X.: Salient object detection with recurrent fully convolutional networks. IEEE TPAMI **41**, 1734–1746 (2018)
28. Wang, T., et al.: Detect globally, refine locally: a novel approach to saliency detection. In: IEEE CVPR, pp. 3127–3135 (2018)
29. Wei, Y., Wen, F., Zhu, W., Sun, J.: Geodesic saliency using background priors. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7574, pp. 29–42. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33712-3_3
30. Yang, C., Zhang, L., Lu, H., Ruan, X., Yang, M.: Saliency detection via graph-based manifold ranking. In: IEEE CVPR, pp. 3166–3173 (2013)
31. Zeng, Y., Lu, H., Zhang, L., Feng, M., Borji, A.: Learning to promote saliency detectors. In: IEEE CVPR, pp. 1644–1653 (2018)
32. Zhang, J., Sclaroff, S., Lin, Z., Shen, X., Price, B., Mech, R.: Minimum barrier salient object detection at 80 FPS. In: IEEE ICCV, pp. 1404–1412 (2015)
33. Zhao, R., Ouyang, W., Li, H., Wang, X.: Saliency detection by multi-context deep learning. In: IEEE CVPR, pp. 1265–1274 (2015)
34. Zhu, W., Liang, S., Wei, Y., Sun, J.: Saliency optimization from robust background detection. In: IEEE CVPR, pp. 2814–2821 (2014)
35. Zhu, Z., Chen, Q., Zhao, Y.: Ensemble dictionary learning for saliency detection. Image Vis. Comput. **32**(3), 180–188 (2014)