iSalGAN - An Improvised Saliency GAN

Kelam Goutam and S. Balasubramanian(✉)

Sri Sathya Sai Institute of Higher Learning, Prashanti Nilayam, Anantapur, India
kelamgoutam.sssihl@gmail.com, sbalasubramanian@sssihl.edu.in
http://sssihl.edu.in/

Abstract. Human visual system (HVS) is naturally attracted to the salient regions that appear distinctly in the foreground of a scene. However, for a machine, automatically detecting the region of saliency is a challenging problem. Recently, a generative model namely Saliency GAN (SalGAN) discriminates if a pixel is salient or not by generating the saliency map given the input image. The generator is guided by a content loss and adversarial loss. However, the generated saliency maps tend to be smooth lacking finer details. We propose an improvised generator called iSalGAN (improvised saliency GAN) that integrates both low-level and high-level features to produce finer saliency maps. Our iSalGAN is guided by a combination of multiple content losses and, the adversarial loss. Our model is trained on MSRA10K dataset and tested on ECSSD and DUT-OMRON datasets. Qualitative and quantitative evaluation of our model shows the superior performance of our model over state-of-the-art methods. Codes will be made publicly available.

Keywords: iSalGAN · Saliency GAN · SalGAN · Generator · Discriminator · Saliency

1 Introduction

The HVS receives about 10^8 to 10^9 bits of information every second. In order to process such huge data in real time, HVS uses its ability to selectively focus on different parts of the scene. Given an image, the nervous system selects part of the scene for further detailed processing, while discarding the rest. It also prioritizes the selected part such that the most relevant parts are processed first. This selection and ordering process is known as selective attention or visual saliency [4].

The visual attention model aims to predict the salient regions of the image. The salient region detection can save computational resources as only the relevant information is processed. It can also be used as a preprocessing step for many other computer vision tasks such as object detection, object recognition etc.

Deep learning (DL) models, particularly convolutional neural networks (CNN) have achieved tremendous success in many of the computer vision tasks such as image classification [5], image segmentation [12] etc. Hence, deployment

R. V. Babu et al. (Eds.): NCVPRIPG 2019, CCIS 1249, pp. 219–228, 2020.
https://doi.org/10.1007/978-981-15-8697-2_20

of DL techniques for saliency prediction is a natural extension. The fully convolutional networks (FCN) being used to predict saliency maps achieves significant improvement over traditional approaches. However, these networks fail to produce sharp saliency maps. The saliency maps produced by these networks miss the fine details and, the boundaries are blurred.

In this work we improvise an adversarial training based architecture called SalGAN [14] to eliminate the blurriness at boundary pixels and produce a sharp saliency map for the given input image. We integrate both low-level and high-level features at the generator to produce low-level, high-level and combined saliency maps. By low-level we mean lower layer features and by high-level we mean higher layer features. The integration of low-level and high-level features has been inspired by [2]. We supervise the saliency maps using a loss function which is a combination of content loss at each level and, the adversarial loss. The discriminator decides the real vs fake between the ground truth saliency map and the combined saliency map produced by the generator. Our method is called as iSalGAN.

2 Related Work

Traditionally, saliency prediction is based on manually engineered features like texture, contrast etc. These methods lacked success as the manually engineered features could not capture the global semantics of the given input image. Presently, with a relatively significant volume of data available, it is a routine work for CNNs to capture global semantics and predict salient regions with higher accuracy than the traditional methods.

An early work in this direction is by Long et al. [12]. Subsequently, Liu et al. [11] designed a neural network consisting of two parts to predict the saliency map. The first subnet, a deep hierarchical saliency network (DHSNet), acts as an encoder network and predicts coarser global features. The coarser global features are then refined using the second subnet, a hierarchical recurrent convolutional neural network (HRCNN), to obtain finer local features. Kümmerer et al. [6] proposed the first transfer learning model for saliency prediction. Their model DeepGaze is a modification of AlexNet architecture [5]. DeepGaze omitted all of the fully connected layers and passed the features of the convolutional layers to a linear model as input to learn the weights. Huang et al. [3] introduced a deep neural network (DNN) model to reduce the semantic gap present between the predicted saliency map and the human's behavior. They redesigned an existing DNN for object recognition and used it for saliency prediction. Pan et al. [15] designed a shallow and a deep convolutional model, trained end-to-end, to detect the salient region in an image. The shallow network is trained from scratch and the deep network is trained using transfer learning.

Different loss functions have been used by different methods mentioned above. The definition of 'best' among them is debatable. To break the continuity of this exploration, instead of tailor making a loss function for the method, Pan et al. [14] proposed a adversarial training based saliency prediction called SalGAN.

Given an input image, the generator generates a saliency map with an aim to fool the discriminator that it is the real saliency map of the given image. Over a period of training guided by binary cross entropy (BCE) loss and adversarial loss, the generator produces accurate saliency maps. However, these saliency maps lack fine quality and are blurred. In this work we improvise SalGAN (iSalGAN) to eliminate the blurriness at boundary pixels and produce a sharp saliency map for the given input image. Our **contributions** are as follows:

- In iSalGAN, we integrate both low-level and high-level features at the generator to produce low-level, high-level and combined saliency maps. In contrast, SalGAN only works with a single layer output.
- In iSalGAN, we supervise these maps using a combination of content loss at each level and, the adversarial loss. In contrast, SalGAN uses only one content loss.
- Unlike VGG-16 used by SalGAN [14] for generator, we use ResNeXt-101. We gain a significant reduction in number of learnable parameters. The reason for this switch is further explained later.
- We compare iSalGAN with SalGAN and other state-of-the-art methods.

3 Proposed Method

Conventionally, in the CNN setting, only the final layers predict the saliency maps, independent of other layers. When an image passes through a neural network, the feature maps are constantly refined by the layers. The final layers use these enriched feature maps to make predictions about the salient objects in the image. Though CNN predict significantly better saliency maps compared to traditional approaches, making predictions independent of other layers does not take multi-scale semantics into consideration. SalGAN too uses a CNN in the generator that does not consider multi-scale semantics.

The proposed improvisation, iSalGAN, leverages on the salient features learned across multiple layers of the network.

3.1 iSalGAN Architecture

iSalGAN consists of a generator and a discriminator. Given an image to the generator, it extracts low-level and high-level features by passing the image through a feature extractor network. It then integrates all the low-level and high-level features respectively. Low-level features attend to fine details while high-level features capture the global semantics. The integrated low-level and high-level features are used to predict intermediate saliency maps respectively. The integrated feature maps are further fused to predict a combined high-resolution saliency map as output. The intermediate saliency maps are used to compute the content loss and the combined saliency map becomes the input to the discriminator for adversarial training. The discriminator attempts to differentiate between the synthesized high-resolution saliency map and the real saliency map which is the ground truth. Figure 1 illustrates the overall architecture of iSalGAN.

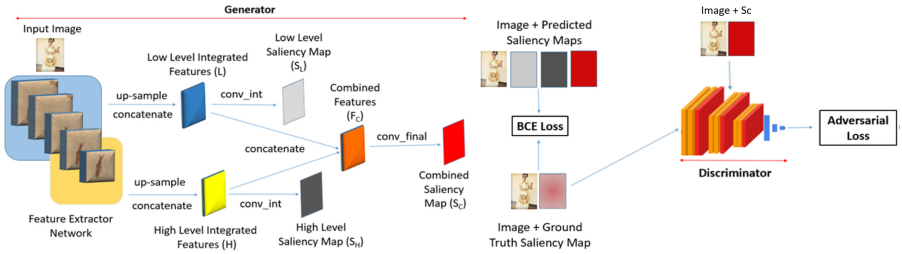


Fig. 1. The overall architecture of iSalGAN (Color figure online)

3.2 Generator

The generator in our iSalGAN network uses ResNeXt-101 [21] as the feature extractor. Given an image to the generator, the ResNeXt model yields a set of feature maps. These feature maps contain low-level as well as high-level semantic information of varying scales. The low-level features and the high-level features are extracted by the shallow layers (grouped in light blue in Fig. 1) and the deep layers (grouped in light yellow in Fig. 1) respectively. These features are up-sampled and concatenated to produce low-level integrated feature map, L (denoted in dark blue in Fig. 1) and high-level integrated feature map, H (denoted in dark yellow in Fig. 1) respectively. The low-level integrated feature, L, and the high-level integrated feature, H, are passed through a shallow convolutional network (denoted as conv_int) to produce low-level saliency map, S_L (denoted in light grey in Fig. 1) and high-level saliency map, S_H (denoted in dark grey in Fig. 1) respectively. The low-level integrated feature, L, and the high-level integrated feature, H, are further combined to produce a richer feature map, F_C (denoted in orange). The combined feature map, F_C, is then passed through another shallow convolutional network (denoted as conv_final) to produce a combined saliency map, S_C (denoted in red in Fig. 1). The generator therefore produces three saliency maps for each input image.

It is to be noted that SalGAN uses VGG-16 [18] as the feature extractor network in the generator. VGG-16 has 138 million learnable parameters. In order to reduce computation overload and memory footprint, SalGAN trades with accuracy by considering only last two groups of convolutional parameters for learning. For other parameters, weights are transfered from VGG-16 pre-trained for ImageNet challenge [16]. Recently, it has been shown that ResNeXt [21] significantly brings down the validation error on ImageNet. A ResNeXt block has varied number of residual paths, each with same topology with significantly less width. This helps in embedding the input into different subspaces thereby able to generalize well across variations. We do not want to trade with accuracy and so we train our iSalGAN model end-to-end. We use ResNeXt-101 that has roughly around 44 million parameters, less than VGG-16 by a factor of 3.

Figures 2 and 3 describe the detailed architecture of the shallow convolutional networks used to generate intermediate and final saliency maps respectively.

Layer	In-channels	Out-channels	Kernel	Activation
Conv1	256	128	3 x 3	PReLU
BatchNorm1	128	-	-	-
Conv2	128	128	3 x 3	PReLU
BatchNorm2	128	-	-	-
Conv3	128	1	1 x 1	sigmoid

Fig. 2. Architecture of conv_int which generates intermediate saliency maps

Layer	In-channels	Out-channels	Kernel	Activation
Conv1	2	128	3 x 3	PReLU
BatchNorm1	128	-	-	-
Conv2	128	128	3 x 3	PReLU
BatchNorm2	128	-	-	-
Conv3	128	1	1 x 1	sigmoid

Fig. 3. Architecture of conv_final which generates final saliency map

3.3 Discriminator

The discriminator network used is same as given in SalGAN [14]. It consists of six convolutional layers with a kernel size of 3×3. A ReLU layer follows each of the convolutional layer, and after every set of two convolutional layers, a maxpool layer follows which reduces the feature size by half. Finally, three fully connected layers follow the convolutional layers. Tanh is used as an activation function for the first two fully connected layers whereas the final fully connected layer uses sigmoid.

4 Training

Our iSalGAN network uses a combination of content loss and adversarial loss. The content loss in our model is computed by combining the losses of the intermediate saliency maps and the final saliency map with respect to the ground truth, respectively. The adversarial loss determines the discriminator's ability to distinguish the combined saliency map, S_C, as real or fake.

4.1 Content Loss

The content loss is defined as:

$$\mathscr{L}_{BCE} = BCE_{S_L} + BCE_{S_H} + BCE_{S_C}$$

where

$$BCE_{S_L} = -\frac{1}{N}\sum_{k=1}^{N}(S^k log({S_L}^k) + (1 - S^k) log(1 - {S_L}^k))$$

$$BCE_{S_H} = -\frac{1}{N}\sum_{k=1}^{N}(S^k log({S_H}^k) + (1 - S^k) log(1 - {S_H}^k))$$

$$BCE_{S_C} = -\frac{1}{N}\sum_{k=1}^{N}(S^k log({S_C}^k) + (1 - S^k) log(1 - {S_C}^k))$$

Here, S^k and ${S_i}^k$, $i = \{L, H, C\}$ represent the probability of the k^{th} pixel being salient in the ground truth and predicted saliency maps respectively and N is the number of pixels in the image. In summary, the content loss is computed by comparing the similarity between the predicted saliency maps with respect to the ground truth saliency map for every pixel.

4.2 Adversarial Loss

The loss function for the discriminator architecture is defined as:

$$\mathscr{L}_{Dis} = L(\mathscr{D}(I, S), 1) + L(\mathscr{D}(I, \tilde{S}), 0)$$

where L denotes BCE loss, the number 1 represents that target belongs to ground truth and 0 represents that it is predicted. $\mathscr{D}(I, \tilde{S})$ represent the probability of fooling the discriminator (i.e. given a predicted saliency map as input, the discriminator classifies it as real). $\mathscr{D}(I, S)$ represent the probability that given a ground truth saliency map, the discriminator predicts it as real. The loss function used in adversarial training is defined as:

$$\mathscr{L} = \alpha \times \mathscr{L}_{BCE} + L(\mathscr{D}(I, \tilde{S}), 1)$$

The loss function $\mathscr{L}$ aids in improving the convergence rate and stability of the adversarial training.
The training of iSalGAN happens in two phases:

1. Pretrain the generator for 15 epochs using only content losses.
2. Subsequently add discriminator and start the adversarial training.

During the adversarial training, the input to the iSalGAN is an RGB image of shape $256 \times 192 \times 3$. Input to the discriminator is an RGBS image of shape $256 \times 192 \times 4$. Generator and the discriminator are trained in alternative iterations. Weight decay is set to 1×10^{-4}. Learning rate is set to 3×10^{-4}. SGD is used as optimizer. Batch size is set to 8. A larger batch size would give better accuracy but due to limitation of resources we worked with batch size of 8. α is set to 5×10^{-3}. The entire network is trained for 120 epochs.

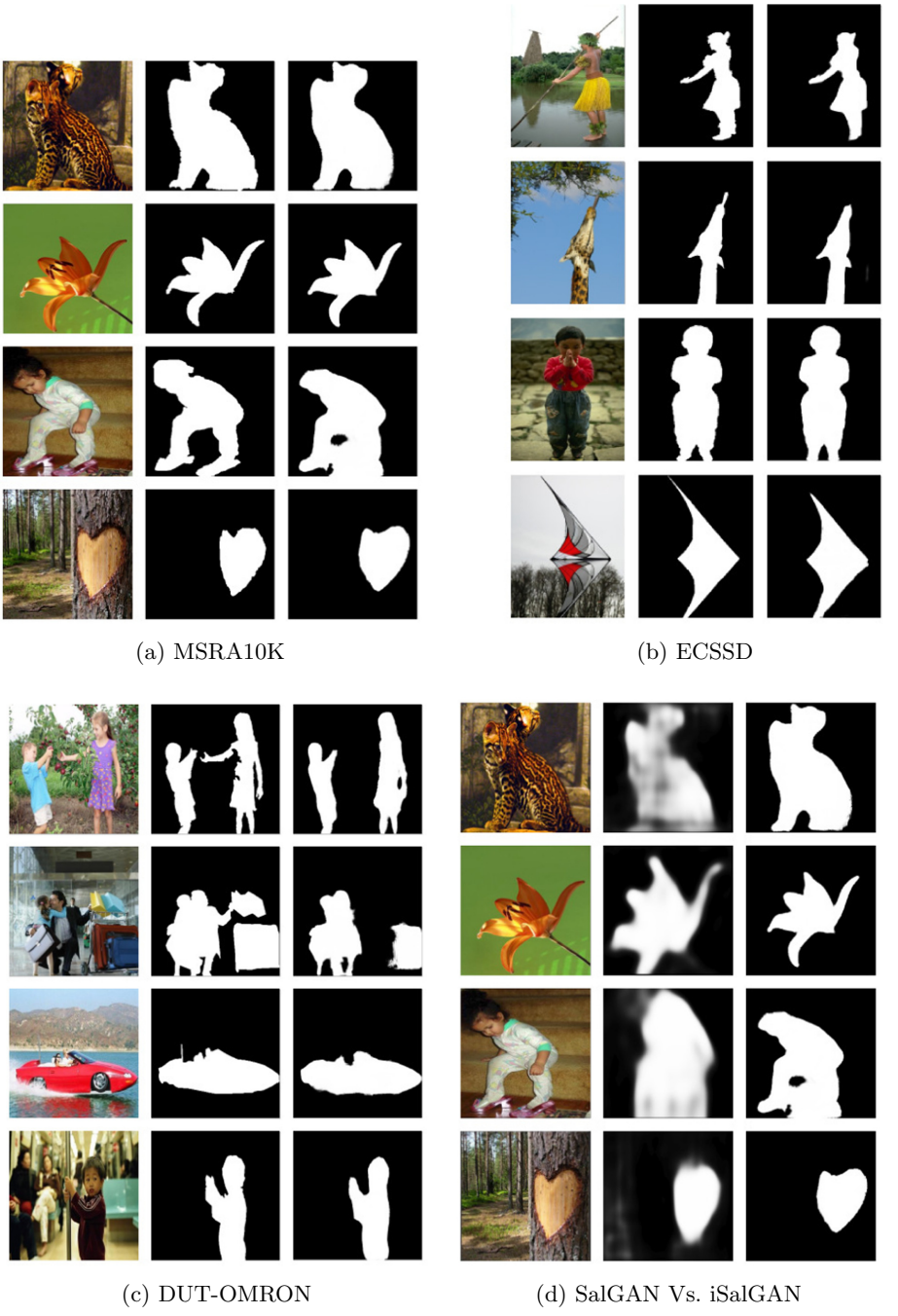
5 Results

In this section, we qualitatively and quantitatively report the results of our iSalGAN model for saliency prediction. The model is trained on MSRA10K dataset [1] and is tested on ECSSD [22] and DUT-OMRON [23] datasets. Parts a, b and c of Fig. 4 depict a sample of results of iSalGAN on MSRA10K, ECSSD and DUT-OMRON datasets. In the above mentioned figures, the first column consists of the query images, the second column consists of the ground truth saliency maps for the corresponding images and the third column shows the predicted saliency maps. Clearly, the results are impressive. Part d of Fig. 4 compares iSalGAN with SalGAN qualitatively. We can clearly emphasize on the sharpness of iSalGAN results over the blurry results produced by SalGAN. Even the minute variations have been reasonably picked up by iSalGAN while SalGAN completely averages them out.

Table 1. Comparison of iSalGAN with the state-of-the-art models for saliency prediction.

	ECSSD		DUT-OMRON	
	F-measure	MAE	F-measure	MAE
BSCA [17]	0.758	0.183	0.616	0.191
MC [24]	0.822	0.106	0.703	0.088
LEGS [19]	0.827	0.118	0.669	0.133
MDF [8]	0.831	0.108	0.694	0.092
ELD [7]	0.867	0.080	0.716	0.091
RFCN [20]	0.898	0.097	0.747	0.095
DS [10]	0.882	0.123	0.745	0.120
DCL [9]	0.898	0.071	0.757	0.080
DHSNet [11]	0.907	0.059	–	–
NLDF [13]	0.905	0.063	0.753	0.080
iSalGAN (ours)	**0.912**	**0.053**	**0.759**	**0.076**

We compared our iSalGAN model with 10 of the state-of-the-art models in literature, using the F-measure and mean absolute error (MAE) metrics. Table 1 shows that iSalGAN outperforms its competitors on both the ECSSD [22] and DUT-OMRON [23] datasets. Further, Table 2 illustrate that iSalGAN outperforms the SalGAN model. With respect to F-measure a significant jump of 8% is observed while the MAE has reduced by a factor of 10. To compare against SalGAN, we trained SalGAN on MSRA10K dataset for 120 epochs. The iSalGAN model is implemented using PyTorch framework. Both the qualitative and quantitative results clearly emphasize the importance of integration of both lower layer and higher layer features and also supervision at both levels.



(a) MSRA10K (b) ECSSD

(c) DUT-OMRON (d) SalGAN Vs. iSalGAN

Fig. 4. Qualitative results of iSalGAN on MSRA10K, ECSSD, DUT-OMRON datasets and SalGAN vs. iSalGAN

Table 2. Quantitative comparison of iSalGAN with the SalGAN model.

Method	MSRA10K	
	F-measure	MAE
SalGAN [14]	0.869	0.103
iSalGAN (ours)	**0.945**	**0.027**

6 Conclusion

The saliency maps generated using the SalGAN architecture have blurred boundaries and using them to segment the salient objects may either add a non-salient part to the segmented object or may ignore some part of the salient object. Such segmentation may affect the accuracy in case of applications like medical image analysis. In order to eliminate the blurriness of the boundary and retain the advantages provided by the SalGAN architecture, we designed an improvised SalGAN called iSalGAN to predict saliency map with clear boundaries. Our iSalGAN model considers both low-level features and high-level feature as equally important. The iSalGAN architecture performed better than 10 of the state-of-the-art models when compared using MAE and F-measure metrics. A future direction would be to extend iSalGAN to predict instance level saliency maps.

Acknowledgements. We would like to dedicate our work to founder chancellor of Sri Sathya Sai Institute of Higher Learning, Bhagawan Sri Sathya Sai Baba.

References

1. 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 (2014)
2. Deng, Z., et al.: R3net: recurrent residual refinement network for saliency detection. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence, pp. 684–690. AAAI Press (2018)
3. Huang, X., Shen, C., Boix, X., Zhao, Q.: Salicon: reducing the semantic gap in saliency prediction by adapting deep neural networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 262–270 (2015)
4. Itti, L., Koch, C., Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. IEEE Trans. Pattern Anal. Mach. Intell. **11**, 1254–1259 (1998)
5. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
6. Kümmerer, M., Theis, L., Bethge, M.: Deep gaze i: boosting saliency prediction with feature maps trained on imagenet. arXiv preprint arXiv:1411.1045 (2014)
7. Lee, G., Tai, Y.W., Kim, J.: Deep saliency with encoded low level distance map and high level features. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 660–668 (2016)

8. 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)
9. Li, G., Yu, Y.: Deep contrast learning for salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 478–487 (2016)
10. Li, X., et al.: Deepsaliency: multi-task deep neural network model for salient object detection. IEEE Trans. Image Process. **25**(8), 3919–3930 (2016)
11. Liu, N., Han, J.: Dhsnet: deep hierarchical saliency network for salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 678–686 (2016)
12. 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)
13. Luo, Z., Mishra, A., Achkar, A., Eichel, J., Li, S., Jodoin, P.M.: Non-local deep features for salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6609–6617 (2017)
14. Pan, J., et al.: Salgan: visual saliency prediction with generative adversarial networks. arXiv preprint arXiv:1701.01081 (2017)
15. Pan, J., Sayrol, E., Giro-i Nieto, X., McGuinness, K., O'Connor, N.E.: Shallow and deep convolutional networks for saliency prediction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 598–606 (2016)
16. Russakovsky, O., et al.: Imagenet large scale visual recognition challenge. Int. J. Comput. Vision **115**(3), 211–252 (2015)
17. Shen, W., Liu, R.: Learning residual images for face attribute manipulation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4030–4038 (2017)
18. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
19. Wang, L., Lu, H., Ruan, X., Yang, M.H.: Deep networks for saliency detection via local estimation and global search. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3183–3192 (2015)
20. Wang, L., Wang, L., Lu, H., Zhang, P., Ruan, X.: Saliency detection with recurrent fully convolutional networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9908, pp. 825–841. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46493-0_50
21. Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1492–1500 (2017)
22. 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)
23. 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)
24. 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)