Saliency Detection by Conditional Generative Adversarial Network

Xiaoxu Cai, Hui Yu*
University of Portsmouth, Portsmouth, United Kingdom
hui.yu@port.ac.uk

ABSTRACT

Detecting salient objects in images has been a fundamental problem in computer vision. In recent years, deep learning has shown its impressive performance in dealing with many kinds of vision tasks. In this paper, we propose a new method to detect salient objects by using Conditional Generative Adversarial Network (GAN). This type of network not only learns the mapping from RGB images to salient regions, but also learns a loss function for training the mapping. To the best of our knowledge, this is the first time that Conditional GAN has been used in salient object detection. We evaluate our saliency detection method on 2 large publicly available datasets with pixel accurate annotations. The experimental results have shown the significant and consistent improvements over the state-of-the-art methods on a challenging dataset, and the testing speed is much faster.

Keywords: saliency detection, deep learning, Generative Adversarial Network, CGAN.

1. INTRODUCTION

Saliency detection devoting to highlight the complete salient objects is one of the fundamental problems, whichhas been drawn much attention in recent years. It has large wide applications in computer vision tasks, such as object recognition [1], image segmentation [2], and person re-identification [3]. During the past few years, we have witnessed significant improvements on these tasks since lots of conventional classic methods [4, 5, 6, 7] and deep learning [8, 9] have been proposed to obtain the salient information.

Accuracy and efficient representation features have been required to estimate the saliency. Most of existing conventional saliency detection methods [4, 5, 6, 7] mainly devote to design the low-level saliency cues about image colour, edge, and texture, or extract the middle-level object information on contour, shape, and spatial context. However, these hand-crafted features cannot make obvious contrast between background and the salient object. This key and challenging issue can be resolved by high-level representation features and have attracted the attention of many researchers. The deep Convolutional Neural Network (CNN) was proposed in [8] to solve the existing problem due to its powerfulness of extracting high-level feature representations [10]. They use parallel CNNs to extract global context and extract local context to model the saliency and refine the saliency respectively. However, this method needs to detect a variety of regions first and then select the salient objects from them. Furthermore, Fully Convolutional Networks (FCN) [9] were proposed to directly create pixel-to-pixel saliency map in an end-to-end manner to improve the efficiency. It seems that the saliency detection task can be regarded as image-to-image translation. Recently, a Conditional Generative Adversarial Net (CGANs) [11] has been proposed and demonstrated its effective performance in day-to-night scenery translation, edge-to-object translation and so on.

In this paper, we concentrate on detecting distinctive regions by using CGAN. GAN [12] was proposed by Goodfellow et al. in 2014. It has already been successfully utilized in many fundamental tasks such as texture generation [13], arbitrary face generation [14], and hand-craft digits generation. Inspired by these, we implement GAN in saliency generation. To our knowledge, this is the first time that GAN has been used in saliency detection. In conventional GANs, G takes a random noise vector to synthesize an image. But in our task, we feed the label to both G and D. *G* takes a RGB image x and a random noise vector z as the inputs, with the goal of generating a salience map y can fool *D*. Specifically, *G* learns a mapping from the input image to a saliency map *l*. The created map *y* is then concatenated with the RGB images to feed to D. At the same time, the ground truth saliency map *l* is also concatenated with RGB image to feed to *D*. *D* tries its best to distinguish the real pair from all the pairs. The creating saliency samples can be been in Figure 1.

This paper makes following contributions:

1) We propose to detect the salient region by using Conditional GAN. We use encoder and decoder structure in G to share the low-level information and U-net to improve the performance in detail.

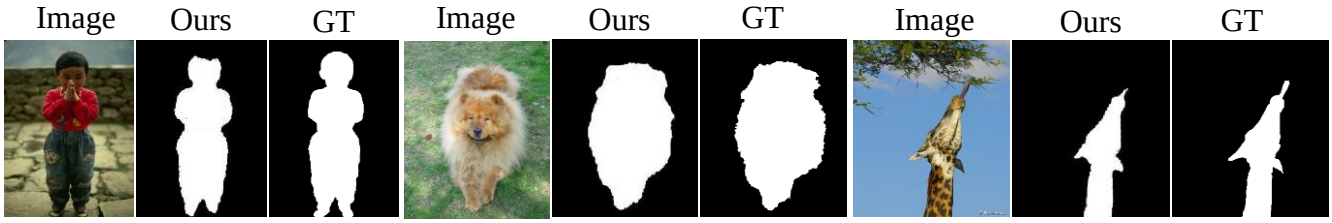

Figure 1. We use conditional GANs to create saliency maps. Maps are similar to Groundtruth.

2) We have achieved the state-of-the-art performance on a challenging dataset.

3) The testing speed is much higher than state-of-the-art deep learning methods.

2. RELATED WORK

2.1 Generative Adversarial Network (GAN)

Goodfellow et al. [12] proposed GAN to learn generative models via an adversarial process. GAN model mainly includes two parts, one is generator which is used to generate images with random noises, and the other one is the discriminator used to distinguish the real image and fake image (generated image). During the adversarial game, the generator improves its ability of generating images closed to real images and the discriminator improves its ability of distinguishing respectively. GAN has been used for image synthesis, image super resolution and so on. However, the network is not stable and sometimes creates some noise images. CNN was applied in both *D* and *G* instead of original generator and discriminator to make the network more stable and effective [15]. Based on this, labeled data were feed to both *G* and *D* to train a supervised model [11]. The similar method has performed well in arbitrary face generation.

2.2 Saliency Detection

During the past few years, image saliency detection has been extensively studied and a variety of methods have been proposed to represent salient maps. These methods can be summarized into conventional methods and deep learning. The former methods [4, 5, 6, 7] try to predict scene locations where a human observer often noticed. Salient object detection [4], [6], [16] aims at highlighting the salient region, which has been shown benefits to a wide range of computer vision applications. Much more detailed reviews of the saliency models can be found in [17]. More recently, deep learning techniques have been introduced to image saliency detection. These methods [8], [18] typically use CNN to examine a variety of region proposals, from which the salient objects are selected. Currently, more and more methods tend to learn in an end-to-end manner and directly generate pixelwise saliency maps via fully convolutional networks (FCN) [9], [19].

Saliency models can be further divided into static and dynamic ones according to their input. In this work, we aim at detecting saliency object regions in static images. In this paper, we propose a deep learning model for detecting salient object regions. The proposed model is effective, yet more computationally efficient compared with existing saliency models. To the best of our knowledge, this paper is the first work to use conditional GAN in detecting salient region.

3. CONDITIONAL GAN FOR SALIENCY DETECTION

3.1 Conditional GAN Architecture Overview

We start with an overview of our Conditional GAN saliency detection model before going into details below. The supervised network is trained with RGB images and salient binary images to learn a mapping or a distribution used to calculate the salient region in an image. Figure 2 shows the workflow.

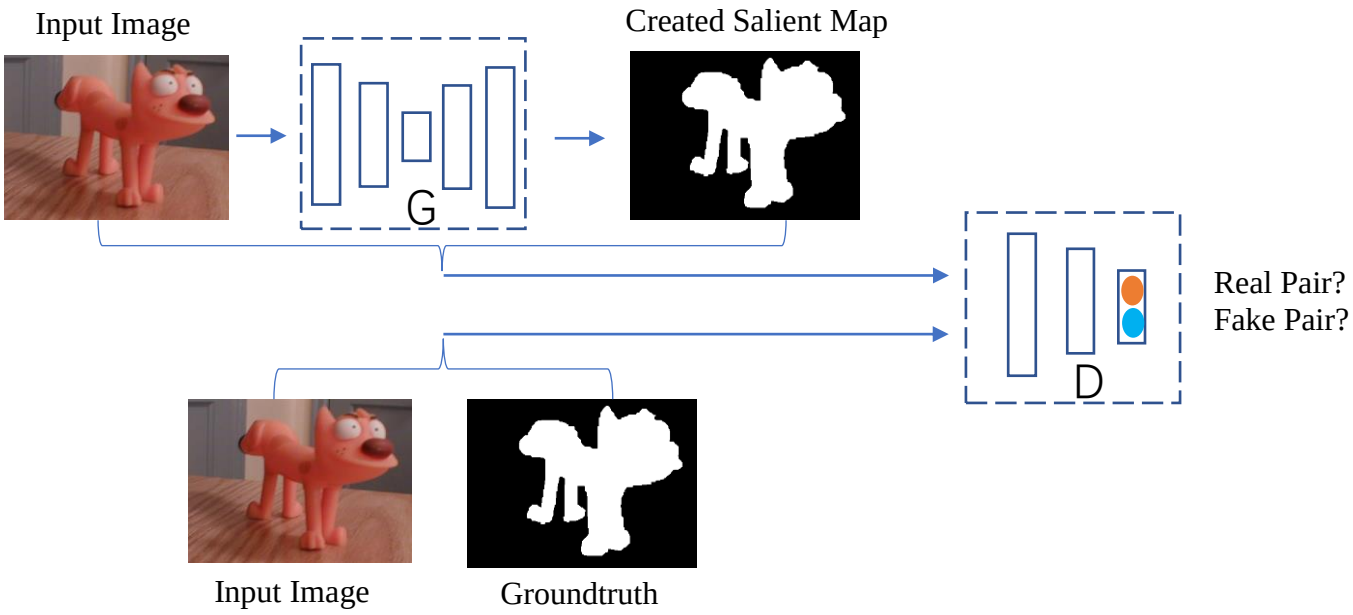

Figure 2. The framework of the training process. We use both encoder and decoder in *G*, it is a good way to share the low-level information between the input and output at the bottleneck. We also add skip connections in *G* following the general shape of a U-net [20] to improve the performance in details. Specifically, we add skip connections between each ith layer and *(n – i)*th layer, where n is the total number of layers. Each skip connection simply concatenates all channels at layer ith with those at layer *(n – i)*th. The structure of *D* is similar to DCGAN [15], using convolutional network instead of *G* in original GANs to be more stable and efficient. Once the saliency model is built,we can get the created salient map by for the new image input to the model.

3.2 Saliency detection using conditional GAN

3.2.1 GAN

GAN mainly includes two parts, one is a generator *G* and the other is a discriminator *D*. It sets up a game *V (G, D)* between *G* and *D*. In this game, the aim of *G* is to create samples *G(z)* to fool *D*, in contrast, the aim of *D* is to distinguish the real images from the real images and the created one *G(z)*. *G* trains itself to improve the ability of creating real-looking samples, *D* trains itself to improve the ability of distinguishing real images, respectively. The game is over when *D* cannot discriminate the created samples are real or fake. The process can be expressed by equation (1)

$$\min \max V(D,G) = E_{x\sim p_{d(x)}}[\log D(x)] + E_{z\sim p_{z(z)}}[\log(1 - D(G(z))] \quad (1)$$

It is proved that this minimax game has a global optimum when the distribution p_g of the synthetic samples and the distribution p_d of the training samples are the same. Under mild conditions (e.g., *G* and *D* have enough capacity), p_g converges to p_d. In practice, it is better for *G* to maximize *log(D(G(z)))* instead of minimizing *log (1- D(G(z)))*. Thus, *G* and *D* are trained to alternatively optimize the following objectives:

$$\max V_D(D,G) = E_{x\sim p_{d(x)}}[\log D(x)] + E_{z\sim p_{z(z)}}[\log(1 - D(G(z))] \quad (2)$$

$$\max V_G(D,G) = E_{z\sim p_{d(z)}}[\log D(G(z))] \quad (3)$$

3.2.2 Conditional GAN

Conditional GAN accepts the text [21], labeled data [22] and images [23] to supervise the training process for creating new samples. Compare to original GANs, labeled data *l* are feed to both *G* and *D*. So the aim of G is to create samples *G(x, z)*to fool *D* and the aim of *D* is to distinguish the real images from the real images and the created one *G(x, z)*, The objective of a conditional GAN can be expressed as:

$$\min \max F(D,G) = E_{x\sim p_{d(x)}}[\log D(x,l)] + E_{z\sim p_{z(z)}}[\log(1 - D(x,G(x,z))] \quad (4)$$

In our task, there is a big difference between image x and labeled data (saliency map) *l*. Actually, our task can be regarded as an image-to-image translation task. Inspired by [11], we concatenate image *x* and groundtruth saliency map *l*

to build an image-label pair *xl*, and the generated saliency map *y* is also concatenated with image *x*, then we get a created image-label pair *xy*. Therefore, the objective in our task can be expressed as:

$$\min\max F(D,G) = E_{x\sim p_{d(x)}}[\log D(xl,xy)] + E_{z\sim p_{z(z)}}[\log(1-D(x,G(x,z))] \quad (5)$$

4. EXPERIMENT AND ANALYSIS

4.1 Datasets

We report our performance on two public benchmark datasets following the method mentioned in [8]. ECSSD [24] is the extended dataset of CSSD containing 1, 000 structurally complex images acquired from the Internet, and the groundtruth masks are annotated by five labelers. PASCAL-S [25] was built on the validation set of the PASCAL VOC 2010 segmentation challenge. It includes 850 natural images with both saliency segmentation groundtruth and eye fixation groundtruth. Saliency groundtruth masks were labeled by 12 subjects. In this paper, we just use saliency segmentation groundtruth to measure our generated salient region.

4.2 Training Process

We use the largest salient object dataset, MSRA10k [26], for the training purpose. This dataset including 10k images and the corresponding binarized groundtruth and covering varied image contents, such as indoor, outdoor, human, animals, vehicles, is widely used in saliency detection. We random select 9000 images to train the model and the rested 1000 images are used for validation. We resize the image to 256*256 to satisfy the requirement of the network, because we use encoder and decoder in *G* and linear transformation in *G* which could not process arbitrary size images. When we get the created salient maps, we resize them again to revert to their original sizes.

For the network setting, we follow the optimization strategy in [15]. The batch size is set to be 1. Adam optimizer is used with a learning rate of 0.0002 and momentum 0.5. The whole training process costs about 100 hours (900k iterations) on a PC with the specs of i7 4.0 GHz CPU, a GTX 1080 GPU, and 8G RAM.

4.3 Performance Comparison

4.3.1 Measures

First, we report quantitative evaluation results on four widely used performance measures: precision, recall, F-score and MAE. For each saliency map, we set threshold from 0 to 255 to generate 256 values of every measurement. All the evaluation results on the dataset are obtained via averaging the measures over saliency maps in the dataset.

Precision measures the percentage of the number of detected salient-object regions inside the ground-truth regions over those of detected salient-object regions, while recall is defined as the proportion of the number of detected salient-object regions inside the ground-truth regions over those of the ground-truth regions, as follows:

$$\text{precision} = \frac{the\ detected\ points\ inside\ groundtruth}{the\ detected\ points} \quad (6)$$

and

$$\text{recall} = \frac{the\ detected\ points\ inside\ groundtruth}{the\ salient\ object\ points\ in\ groundtruth} \quad (7)$$

As neither precision nor recall considers the true negative saliency assignments, the mean absolute error (MAE) is also introduced as a complementary measure. MAE is defined as the average per-pixel difference between an estimated saliency map *S* and its corresponding ground truth *G*. Here, *S* and *G* are normalized to the interval [0, 1]. MAE is computed as:

$$MAE = \sum_{i=1}^{h*w} \frac{S(xi)-G(xi)}{h*w} \quad (8)$$

The parameter *h* and *w* refer to the height and width of the input frame image respectively.

The F-measure score is the overall performance measurement computed by the weighted harmonic of precision and recall:

$$F - measure = \frac{(1+\beta^2)\times precision\times recall}{\beta^2\times precision+recall} \quad (9)$$

We set $\beta^2 = 0.3$ to weigh precision higher than recall as suggested in [27]. All the results are showed in Table1.

Table 1. Four measures on ECSSD and PACSAL-S datasets.

Measures	Precision	Recall	MAE	F-measure Score
ECSSD	0.7480	0.8118	0.1153	0.7644
PASCAL-S	0.7065	0.7471	0.1665	0.7151

4.3.2 Comparison

To evaluate the quality of the proposed approach, we provide quantitative comparison for performance of the proposed method against several top-performing alternatives: Graph-based visual saliency(GBVS) [28], saliency filters (SF) [6], global contrast (GC) [4], contextual emergence of object saliency(CEOS) [29], and principal component analysis (PCA)[30], graph -based manifold ranking (GBMR) [31], hierarchical saliency (HS) [24], discriminative regional feature integration (DRFI) [32], and Multi-context deep learning (MCDL) [8]. Figure 3 shows visual comparison of several mentioned methods. In addition, as mentioned above, F- measure score is the overall performance measurement, so we conduct the comparison on this measure. The comparison results can be seen in Table2.

Table 2. The F-measure scores of benchmarking approaches on 2 datasets.

Methods	GBVS	SF	GC	CEOS	PCAS	GBMR	HS	DRFI	MCDL	**OURS**
ECSSD	0.5528	0.5448	0.5821	0.6465	0.5800	0.6570	0.6391	0.6909	0.7322	**0.7644**
PASCAL-S	0.5929	0.5740	0.6184	0.6557	0.6332	0.7055	0.6819	0.7477	0.7940	**0.7151**

Table 2 shows the performance of our method and other nine state-of-the-art methods using these two datasets. As shown in Table 2, our method outperforms the state-of-the-art methods on the ECSSD dataset [24], whileit is not the best one in PASAL-S dataset, a little lower than both DRFI and MCDL. The PASAL-S includes several 2-object and multi-object images, however, the MSRA10k dataset which we use to train the model doesn't include such kind of images. The network has strong learning ability, but it is hard to map the object that it hasn't seen before. As we can see from Fig 3, our method gets a whole contour of salient objects. Overall, our network shows an improvement over the current state-of-the-art across challenging ECSSD dataset. In addition, detecting a saliency map with 256*256 pixels by using our model only takes about 0.25s, much lower than that mentioned in [8, 9].

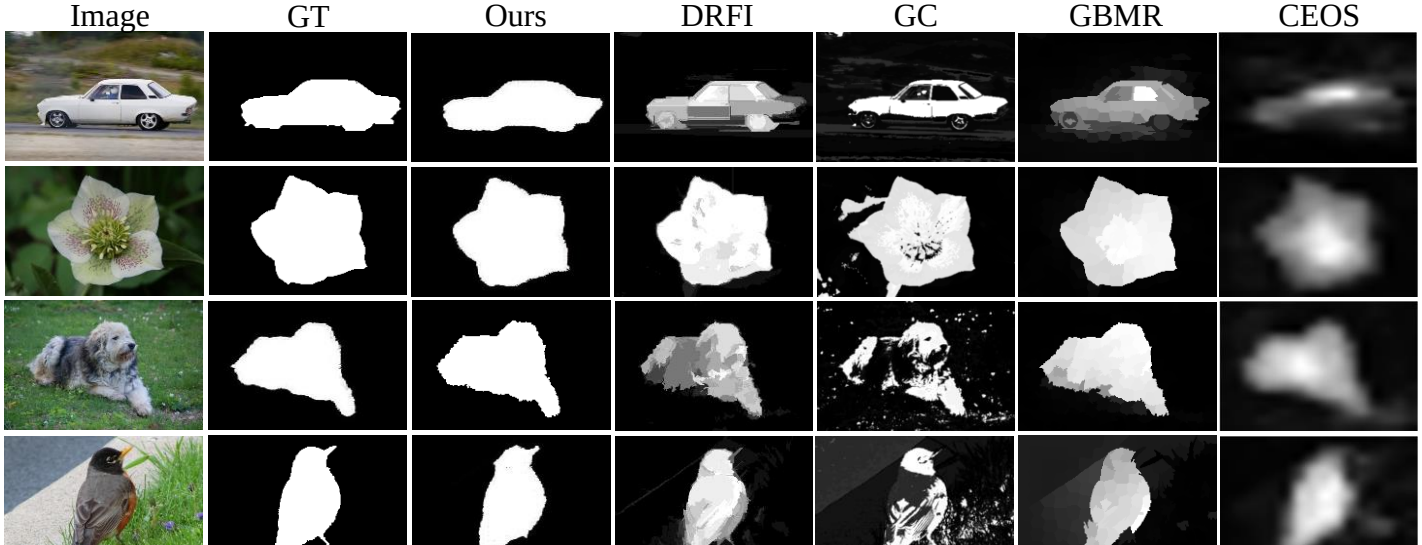

Figure 3. Visual comparison of different methods.

5. CONCLUSION

We have proposed to create a saliency map by using conditional GANs. We use a encoder-decoder structure in G instead of original G to improve the mapping ability between RGB images and saliency binarized maps, U-net has also been used in this process. The Experiment using the F-measure score on 2 large public available datasets shows that our results are better than some previous best results. We have achieved a very good performance in challenging dataset ECSSD. Additionally, our model is efficient, creating 4 saliency maps in 1s on a GPU. In future, we will modify the network to deal with the objects that it hasn't seen in the training process.

Acknowledgement

REFERENCES

[1] Rutishauser, Ueli, Dirk Walther, Christof Koch, and Pietro Perona. "Is bottom-up attention useful for object recognition?." In Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proc IEEE 2, II-II. IEEE, (2004).

[2] Donoser, Michael, Martin Urschler, Martin Hirzer, and Horst Bischof. "Saliency driven total variation segmentation." In Computer Vision, 2009 IEEE 12th International Conference on, 817-824, (2009).

[3] Zhao, Rui, Wanli Ouyang, and Xiaogang Wang. "Person re-identification by salience matching." Proc IEEE, 2528-2535, (2013).

[4] Cheng, Ming-Ming, Jonathan Warrell, Wen-Yan Lin, Shuai Zheng, Vibhav Vineet, and Nigel Crook. "Efficient salient region detection with soft image abstraction." Proc IEEE, 1529-1536, (2013).

[5] Jian, Muwei, Kin-Man Lam, Junyu Dong, and Linlin Shen. "Visual-patch-attention-aware saliency detection." IEEE Trans. Cybernetics 45, 1575-1586, (2015).

[6] Perazzi, Federico, Philipp Krähenbühl, Yael Pritch, and Alexander Hornung. "Saliency filters: Contrast based filtering for salient region detection." Proc IEEE, 33-740, (2012).

[7] Qiang Qi, Muwei Jian, Yilong Yin, JunyuDong, Wenyin Zhang, Hui Yu, "saliency detection using texture and local cuses", CCCV 2017, TianJin, China, (2017).

[8] Zhao, Rui, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. "Saliency detection by multi-context deep learning." Proc IEEE, 1265-1274, (2015).

[9] Wang, Wenguan, Jianbing Shen, and Ling Shao. "Deep Learning For Video Saliency Detection." arXiv preprint arXiv:1702.00871, (2017).

[10] LeCun, Yann, Bernhard Boser, John S. Denker, Donnie Henderson, Richard E. Howard, Wayne Hubbard, and Lawrence D. Jackel. "Backpropagation applied to handwritten zip code recognition." Neural Computation 14, 541-551, (1989).

[11] Isola, Phillip, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. "Image-to-image translation with conditional adversarial networks." arXiv preprint arXiv:1611.07004, (2016).

[12] Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial nets." In Advances in neural information processing systems, 2672-2680, (2014).

[13] Gan, Yanhai, Huifang Chi, Ying Gao, Jun Liu, Guoqiang Zhong, and Junyu Dong. "Perception Driven Texture Generation." arXiv preprint arXiv:1703.09784, (2017).

[14] Tran, Luan, Xi Yin, and Xiaoming Liu. "Disentangled representation learning gan for pose-invariant face recognition." Proc IEEE 45, 7, (2017).

[15] Radford, Alec, Luke Metz, and Soumith Chintala. "Unsupervised representation learning with deep convolutional generative adversarial networks." arXiv preprint arXiv:1511.06434, (2015)

[16] Wang, Wenguan, Jianbing Shen, Ling Shao, and Fatih Porikli. "Correspondence driven saliency transfer." IEEE Trans. Image Proc. **25**, 5025-5034, (2016).

[17] Borji, Ali, Laurent Itti, J. Liu, P. Musialski, and P. Wonka. "State-of-the-art in visual attention modeling." IEEE Trans. Pattern Analysis and Machine Intelligence. **35**, 185-207, (2013).

[18] Li, Guanbin, and Yizhou Yu. "Visual saliency based on multiscale deep features." Proc IEEE, 5455-5463, (2015).

[19] Wang, Linzhao, Lijun Wang, Huchuan Lu, Pingping Zhang, and Xiang Ruan. "Saliency detection with recurrent fully convolutional networks." In European Conference on Computer Vision, 825-841, (2016).

[20] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." In International Conference on Medical Image Computing and Computer-Assisted Intervention, 234-241, (2015).

[21] Reed, Scott, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. "Generative adversarial text to image synthesis." arXiv preprint arXiv:1605.05396, (2016).

[22] Zhou, Zhiming, Shu Rong, Han Cai, Weinan Zhang, Yong Yu, and Jun Wang. "Generative Adversarial Nets with Labeled Data by Activation Maximization." arXiv preprint arXiv:1703.02000, (2017).

[23] Wang, Xiaolong, and Abhinav Gupta. "Generative image modeling using style and structure adversarial networks." In European Conference on Computer Vision, 318-335, (2016).

[24] Yan, Qiong, Li Xu, Jianping Shi, and Jiaya Jia. "Hierarchical saliency detection." Proc IEEE, 1155-1162, (2013).

[25] Li, Yin, Xiaodi Hou, Christof Koch, James M. Rehg, and Alan L. Yuille. "The secrets of salient object segmentation." Proc IEEE, 280-287, (2014).

[26] Cheng, Ming-Ming, Niloy J. Mitra, Xiaolei Huang, Philip HS Torr, and Shi-Min Hu. "Global contrast based salient region detection." IEEE Trans. Pattern Analysis and Machine Intelligence. **37**, 569-582, (2015).

[27] Vig, Eleonora, Michael Dorr, and David Cox. "Large-scale optimization of hierarchical features for saliency prediction in natural images." Proc IEEE, 2798-2805, (2014).

[28] Harel, Jonathan, Christof Koch, and Pietro Perona. "Graph-based visual saliency." In Advances in neural information processing systems, 545-552, (2007).

[29] Mairon, Rotem, and Ohad Ben-Shahar. "A closer look at context: From coxels to the contextual emergence of object saliency." In European Conference on Computer Vision, 708-724, (2014).

[30] Margolin, Ran, Ayellet Tal, and Lihi Zelnik-Manor. "What makes a patch distinct?." Proc IEEE, 1139-1146, (2013).

[31] Yang, Chuan, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. "Saliency detection via graph-based manifold ranking." Proc IEEE, 3166-3173, (2013).

[32] Jiang, Huaizu, Jingdong Wang, Zejian Yuan, Yang Wu, Nanning Zheng, and Shipeng Li. "Salient object detection: A discriminative regional feature integration approach." Proc IEEE, 2083-2090, (2013).