J. Vis. Commun. Image R. 67 (2020) 102761

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Saliency detection using adversarial learning networks ☆

Yong Wu [a,b], Zhi Liu [a,b,*], Xiaofei Zhou [c]

[a] Shanghai Institute for Advanced Communication and Data Science, Shanghai University, Shanghai 200444, China
[b] School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China
[c] Institute of Information and Control, Hangzhou Dianzi University, Hangzhou 310018, China

ARTICLE INFO

Article history:
Received 20 May 2019
Revised 10 September 2019
Accepted 11 January 2020
Available online 18 January 2020

Keywords:
Adversarial learning
Generator
discriminator
Saliency detection
Feedback information

ABSTRACT

This paper proposes a novel model for saliency detection using the adversarial learning networks, in which the generator is used to generate the saliency map and the discriminator is deployed to guide the training process of overall network. Concretely, the training procedure of our model consists of three steps including the training of generator, the training of discriminator, and the training throughout the overall network. The key point of training process lies in the discriminator, which is designed to provide the feedback information for the acceleration of the generator and the refinement of saliency map. Therefore, during the training stage of overall network, the output of the generator, *i.e.* the coarse saliency map, is fed into the discriminator, yielding the corresponding feedback information. Following this way, we can obtain the final generator with a higher performance. For testing, the obtained generator is employed to perform saliency detection. Extensive experiments on four challenging saliency detection datasets show that our model not only achieves the favorable performance against the state-of-the-art saliency models, but also possesses the faster convergence speed when training the proposed model.

1. Introduction

Saliency detection aims at highlighting the most visually distinctive objects or areas in an image, and has been applied in a wide range of applications such as visual tracking [1,2], image captioning [3], image segmentation [4–6], scene classification [7], content-aware image editing [8], salient object detection [9–11], and weakly supervised learning [12–14], to name a few. In recent years, numerous efforts have been proposed, and especially, the convolutional neural networks (CNN) based models [15–24] have pushed forward the progress of saliency detection significantly. However, the performances of these models degrade seriously when dealing with some complicated scenes, such as cluttered background, low contrasts between salient objects and background, and so on. Besides, the recent success of generative adversarial networks (GANs) effectively improved unsupervised and supervised learning in various tasks [25,26]. A classical GAN consists of two sub-networks, namely the generator and the discriminator. The generator captures the data distribution, *e.g.*, a natural image, and the discriminator aims to estimate the probability that a sample comes from the true distribution rather than the generator. The overall network plays a max-min game in the training process. Therefore, it is a natural idea that GAN can be employed to perform saliency detection [27].

Different from the conventional framework of GANs, the adversarial learning is exploited for saliency detection in different ways. Firstly, the aim of saliency detection is to predict a saliency map from an original image rather than generating an image from random noise. Thus, the input to generator is the original image. Secondly, there is no ground truth to compare while a conventional GAN generates a realistic image. Ji et al. [27] proposed a saliency detection model by exploiting conditional adversarial network under the cGAN framework. In this paper, we propose a novel saliency detection model using adversarial learning networks, in which the generator is utilized for saliency detection and the discriminator is used as an adjustment module to accelerate and refine the training of our model. During the training, the generator in our model first produces a coarse saliency map, and then, we train the discriminator in our model. Finally, we train the overall network, in which the weights of the generator are updated by keeping the weights of the discriminator unchanged and backpropagating errors through the discriminator.

☆ This paper has been recommended for acceptance by Zicheng Liu.

* Corresponding author at: School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China.

E-mail address: liuzhisjtu@163.com (Z. Liu).

https://doi.org/10.1016/j.jvcir.2020.102761

Overall, the main contributions of this work can be summarized as follows:

- We propose a novel saliency model using the adversarial learning networks, in which the generator is deployed to perform saliency detection and the discriminator is exploited to promote the model training. Note that a recent model [27] also exploits GAN for image saliency detection, the proposed model is considerably different from [27] in the complete framework of generator, loss function and contribution of discriminator to saliency detection.
- The proposed model is simple yet very effective, and with the faster convergence speed of model training. Extensive experiments on four challenging datasets demonstrate that the proposed model achieves the favorable performance against the state-of-the-art saliency models.

2. Related work

In this section, we give a brief introduction of the state-of-the-art saliency models, and the applications of GANs.

2.1. Saliency detection

Before deep learning revolution, the early efforts on saliency detection mainly focus on extracting low-level visual features by handcrafting, such as center bias and contrast prior [28,29]. In [30], Zou et al. proposed a surroundedness-based multiscale saliency method based on the Gestalt principles for figure-ground segregation. Xu et al. [31] proposed a novel model to measure pixel-level saliency, in which the region-based saliency detection is performed by incorporating a multi-scale segmentation technique. In contrast, with the high-level semantic information generated by convolutional neural networks, the performance of saliency detection has been pushed forward significantly.

Recently, deep learning is more and more important with regard to saliency detection. He et al. [32] utilized CNN to learn super pixel-wise feature representations. Li et al. [33] employed CNN to extract multiscale features for saliency prediction, and Zhao et al. [16] proposed a multi-context deep learning architecture, which uses two pathways for capturing local and global context from two superpixel-centered windows of different sizes. They both utilized fully convolutional neural networks (FCNs) [34,35] to extract high-level semantic cues, to improve saliency detection performance, particularly for images with cluttered background. In [17,36], the short skip connections of FCNs are proposed to combine high-level semantic cues with low-level details that can highlight the features of salient objects clearly. Wang et al. [19] aggregated multi-level features into multiple resolutions, and further refined the multiple aggregated features in a top-down manner before fusion. The recurrent structure can help refine the saliency map by iteratively integrating contextual cues. In [24], Kuen et al. used a convolutional-deconvolutional network to generate the saliency map, and then exploited a recurrent network to further refine the saliency map. Wang et al. [22] adopted the predicted coarse saliency map as the prior, which can refine the saliency map by correcting its previous errors. Wang et al. [18] presented a global recurrent localization network to purify the convolutional features for localizing salient objects accurately. Although the above mentioned models achieve the better performance, the training of these models is time-consuming.

2.2. Generative adversarial networks

Generative adversarial networks (GANs) [37] are usually used to generate images with real statistical distribution. The main idea is that GANs must fit two parameter functions (generator and discriminator) simultaneously. The first function, called generator, is trained to transform samples from random distributions (such as normal distribution) to complicated distributions (*e.g.* natural images). The second function, called discriminator, is trained to differentiate the real sample and the generated fake distribution. The GANs have been successfully applied to many areas such as super-resolution [38], image inpainting [39–41], image translation [42] and facial attribute manipulation [43]. Different from the previous GAN based applications, our work exploits the adversarial networks to facilitate the training process for saliency detection. The models of classical GANs run into trouble to deal with saliency detection task, since it is hard to generate a relevant saliency map of a natural image from random noise. Therefore, we choose the natural image as the input to the generator in our model instead of random noise. Besides, in our model, the generator predicts the coarse saliency map, the discriminator learns the relevance between the input image and its ground truth, which serves as a feedback signal to the generator, and then our model automatically learns to refine the previous coarse saliency map.

3. The proposed model

In this section, we illustrate the architecture of our model for saliency detection in Fig. 1. We first describe the overall architecture of our model including generator and discriminator in Section 3.1, and then introduce the adversarial loss function of our model in Section 3.2. Finally, we detail the training strategy of our model in Section 3.3. As shown in Fig. 1, the main processes of training are as follows. First, we obtain the coarse saliency map using the generator. Then, the coarse saliency map is refined by utilizing the adversarial learning loss with the help of the well-trained discriminator. Finally, the generator can generate the refined saliency map.

3.1. Overall architecture

As shown in Fig. 1, our model consists of two competing convolutional neural networks, *i.e.* generator and discriminator. The generator G in our model is employed to generate the saliency map, and the discriminator D in our model aims at distinguishing the saliency map predicted by the generator from the ground truth.

Generator. The generator adopts the encoder-decoder structure, as shown in Fig. 2. Concretely, for the encoder part, we employ the ResNeXt-101 [44] as the backbone, and correspondingly, discard the last pooling layer and the fully connected layer. In the training phase, the weights of the encoder part are initialized with the ResNeXt-101 trained on the ImageNet. In addition, we deploy the atrous spatial pyramid pooling (ASPP) in the last two layers of ResNeXt-101 to output the two feature maps as shown in the orange box of Fig. 2. Then the two feature maps are concatenated and fed to the two convolution layers with 3×3 kernel and a PReLU activations layer, to obtain the integrated feature map with 256 channels and spatial size of 18×18. For the decoder part, we use two up-sampling layers following the integrated feature map of the encoder and continue to use an up-sampling layer with the binary cross entropy (BCE) loss, and finally the output saliency map has the same spatial resolution as that of the input image.

Discriminator. Fig. 3 shows the architecture of the discriminator D in our model. Concretely, our discriminator consists of 5 convolutional layers with the kernel size of 3×3. Each convolutional layer is followed by a ReLU layer and a max-pooling layer, and the fully connected layer is with 100 dimensions. Finally, the sigmoid activation function is applied to the last layer, in order to

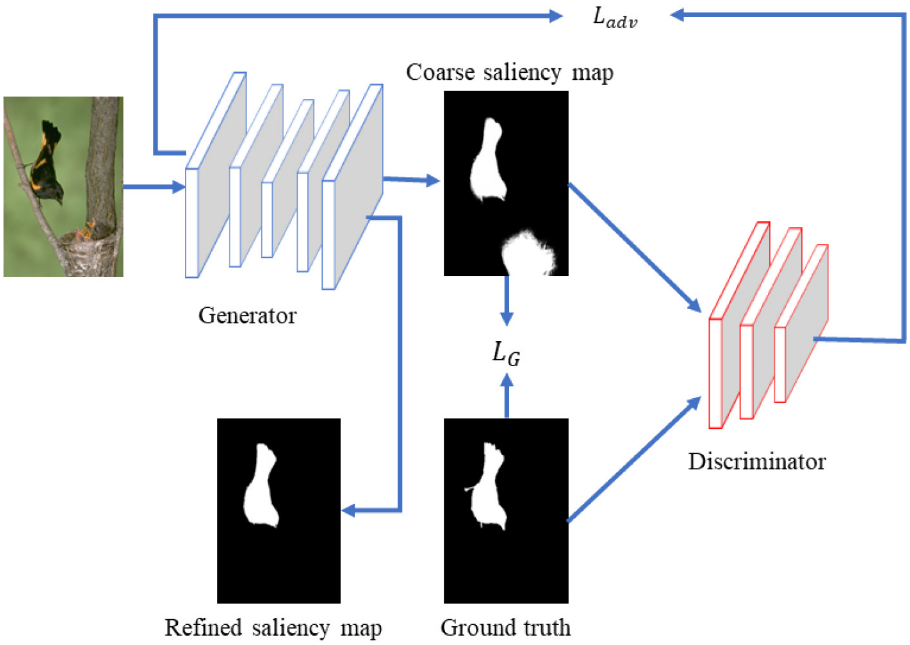

Fig. 1. Schematic illustration of our model.

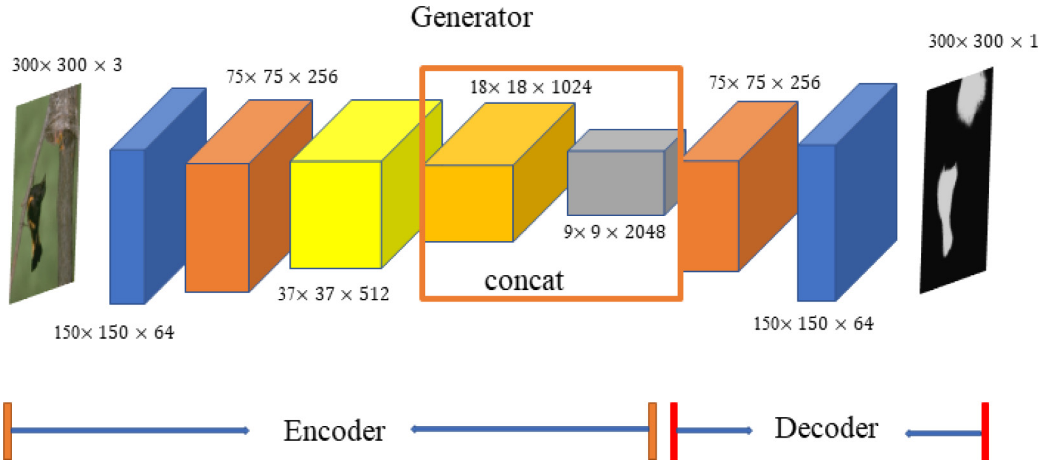

Fig. 2. The architecture of the generator G in our model. The feature maps of last two layers (within the orange box) are concatenated to obtain the integrated feature map. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

obtain a score to identify whether the generated saliency map is from the generator G or from the ground truth.

3.2. Loss function

We give a detailed description for the loss function of our model in the following.

Generator loss. The generator loss of our model is computed in a pixel-wise manner, in which each pixel's value of the predicted saliency map from the generator is compared with the corresponding ground truth. Given an image $\boldsymbol{X}_t$ with a size of $H \times W \times 3$, we denote the predicted saliency map by $\boldsymbol{S}(\boldsymbol{X}_t)$ with a size of $H \times W \times 1$. Then, we can compute the cross-entropy loss between the ground truth $\boldsymbol{Y}_t$ and the predicted saliency map $\boldsymbol{S}(\boldsymbol{X}_t)$. The loss of generator is defined as

$$L_G = -\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{Y_t}log[\boldsymbol{S}(\boldsymbol{X_t})] + (1-\boldsymbol{Y_t})log[1-\boldsymbol{S}(\boldsymbol{X_t})], \quad (1)$$

where T is the total number of training samples.

Discriminator loss. To train the discriminator network, we define the discriminator loss as follows:

$$L_D = \frac{1}{T}\sum_{t=1}^{T}L_{BCE}[D(\boldsymbol{X_t}, \boldsymbol{Y_t}), 1], \quad (2)$$

where L_{BCE} is the binary cross entropy loss. $D(\boldsymbol{X_t}, \boldsymbol{Y_t})$ represents the output of the discriminator, and 1 denotes that the true sample comes from the ground truth.

Adversarial loss. Notably, when updating the weights of the generator, we find that the discriminator is helpful to improve the quality of the predicted saliency map and to accelerate the convergence speed of model training. Therefore, our adversarial loss consists of the generator loss and the output of the discriminator, and the final loss function of adversarial training can be formulated as follows:

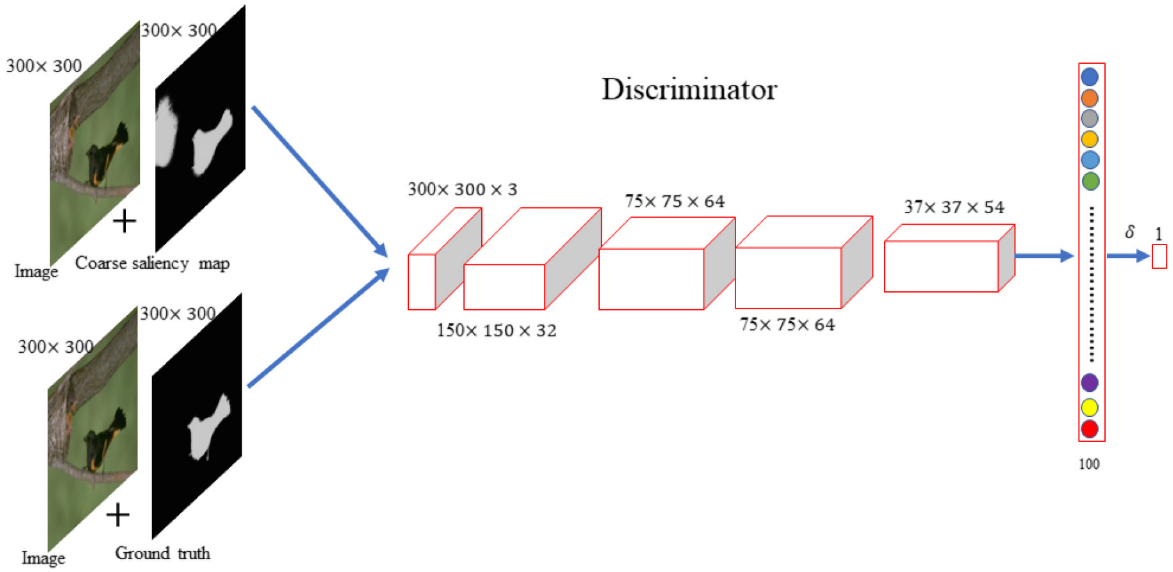

Fig. 3. The architecture of the discriminator. The coarse saliency map generated by the generator and the corresponding ground truth are respectively combined with the RGB image to pass to the discriminator.

$$L_{adv} = L_G + \alpha \frac{1}{T}\sum_{t=1}^{T} logD[\boldsymbol{X_t}, \boldsymbol{S}(\boldsymbol{X_t})], \tag{3}$$

where α is empirically set to -0.2.

3.3. Training strategy

Different from the classical GANs, in this paper, we propose a novel saliency detection model using adversarial learning networks, in which the generator is utilized for saliency detection and the discriminator is used as an adjustment module to accelerate and refine the training of our model. The loss function of the proposed model can be found in Section 3.2. To optimize the adversarial loss defined in Eq. (3), our training strategy contains the following three steps.

Step 1. We first train the generator G in our model for 2,000 iterations with a batch size of 8 using the MSRA10K dataset [45] with the generator loss L_G, and save the trained model.
Step 2. Then, we train the discriminator D in our model by optimizing the discriminator loss L_D. Specifically, the ground truth and the original image are concatenated, and then fed to the discriminator D. We validate our model using the DUT-OMRON dataset every 2000 iterations during the training of discriminator, in which we find that the proposed model convergences when the number of iterations reaches 10 K, then we stop training and keep the weights of the discriminator unchanged for the next stage. The discriminator learns the relevance between the original image and the ground truth.
Step 3. Lastly, we train the overall network on the basis of the above two steps. The generator and the discriminator are jointly trained by optimizing L_{adv} in Eq. (3). As shown in Fig. 1, the coarse saliency map generated by the generator is combined with the original image, and treated as the input to the discriminator. Besides, the parameters of the discriminator are kept unchanged during this step. We stop the training after 3,000 iterations. To show our training procedure simply, we take that into the pseudo code which is shown in **Algorithm 1**.

Algorithm 1. Pseudo Code of Training.

```
for iter1 ← 1–2000 do
    We train the generator for 2000 iterations based on
    the MSRA10K dataset using Eq. (1), and then save the
    model.
end for
for iter2 ← 1–10,000 do
    The RGB images with the corresponding ground
    truths are used as the input to discriminator. We
    train the discriminator using Eq. (2), and we validate
    our model per-2,000 iterations using the DUT-OMRON
    dataset.
end for
for iter1 ← 2001–5000 do
    We train the generator and discriminator using Eq. (3)
    simultaneously.
end for
```

Implementation details. We implemented our model based on Python 3 with the well-known PyTorch framework. We run our model on a single NVIDIA Titan Xp GPU (12 GB memory). We used the pre-trained ResNeXt101 on ImageNet to initialize the generator, and randomly initialized the discriminator with a Gaussian distribution. In the training phase, (1) we use the stochastic gradient descent (SGD) to train the generator with the momentum of 0.9, the weight decay of 0.0005 and the learning rate of 0.001; (2) for the discriminator, we also use the SGD to train the discriminator with the learning rate of 0.0002. For the testing phase, we employ the well-trained generator in our model to predict saliency maps as shown in Fig. 4. Our model can generate the saliency map with the same size as that of the input image. We apply the fully connected conditional random field (CRF) [46] to refine the spatial coherence of the saliency map and the detail of post-processing can be found in Section 3.4.

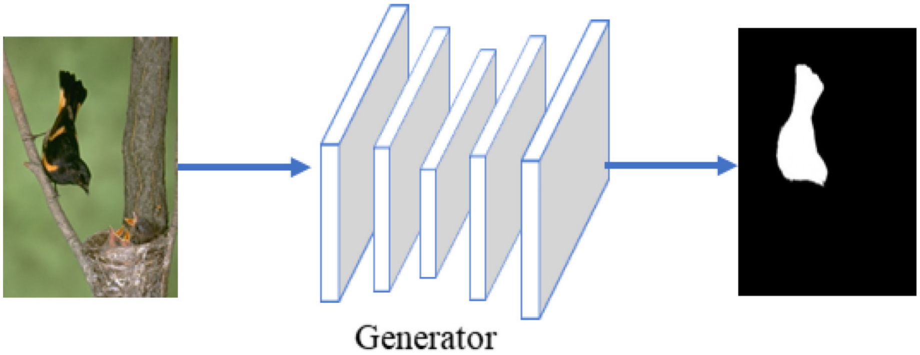

Fig. 4. Overview of the testing phase. We discard the discriminator like the framework of GAN.

3.4. Fully connected CRF

Recent works [47,17] have shown that fully connected conditional random fields (CRFs) can further enhance the performance via the elevation of spatial coherence. The energy function of CRF model is defined as:

$$E(x) = \left[\sum_i \psi_u(x_i)\right] + \left[\sum_{i<j} \psi_p(x_i, x_j)\right], \tag{4}$$

where x denotes a binary label assignment for all pixels, $\psi(x_i)$ is the unary potential, which can be obtained from CNNs. The pair-wise energy $\psi_p(x_i, x_j)$ can maximize the label consistency between adjacent pixels. The pairwise potential is defined as:

$$\begin{aligned}\psi_p(x_i, x_j) = \mu(x_i, x_j)[\omega_1 exp(-\frac{||p_i - p_j||^2}{2\sigma_\alpha^2} - \frac{||I_i - I_j||^2}{2\sigma_\beta^2}) \\ + \omega_2 exp(-\frac{||p_i - p_j||^2}{2\sigma_\gamma^2})],\end{aligned} \tag{5}$$

where $\mu(x_i, x_j) = 1$ if $x_i \neq x_j$, and 0 otherwise. I_i and p_i represent the color vectors and position coordinates of pixel i, respectively. $\psi(x_i, x_j)$ involves two kernels, in which the first kernel pushes nearby pixels with similar colors to take similar saliency scores, and the second kernel removes small isolated regions. All parameters are set according to [46]. To validate the effects of CRF qualitatively, we make a comparison in Fig. 5. Concretely, we define the saliency map generated by the proposed model with and without CRF as 'with CRF' and 'without CRF', respectively. It can be seen that 'with CRF' shown in Fig. 5(d) is the closest one to ground truth (GT), and it preserves boundaries more clearly.

4. Experimental results

4.1. Datasets

We evaluate our model on four benchmark datasets, including HKU-IS [33], ECSSD [48], PACAL-S [49], and DUTS [50]. These datasets contain a variety of images and pixel-level annotations of salient objects. **HKU-IS** is a massive dataset which contains 4,447 challenging images. Most of images have low contrasts and multiple salient objects with overlapping. **ECSSD** is also a challenging dataset which has 1,000 complex and semantically meaningful natural images. **PACAL-S** is an extremely challenging dataset including 850 images with multiple objects. **DUTS** is the largest saliency detection dataset released so far. The DUTS dataset includes 10,553 images for training and 5,019 testing images.

4.2. Evaluation metrics

We use three evaluation metrics, including average F-measure, precision-recall (PR) curve and mean absolute error (MAE) to evaluate the performance of our model and the state-of-the-art saliency models. The PR curve is an important metric to measure the performance of saliency model. Given a saliency map normalized into the range of [0, 255], we calculate the precision value and recall value by varying the integer threshold from 0 to 255. After that, the PR curve is plotted with recall as x-coordinate and precision as y-coordinate. In addition, we exploit the average F-measure value to measure the performance of saliency model. We first obtain the binary object mask using [51] on the saliency map, then the precision and the recall are calculated by comparing the binary object mask with the ground truth, and finally the F-measure value

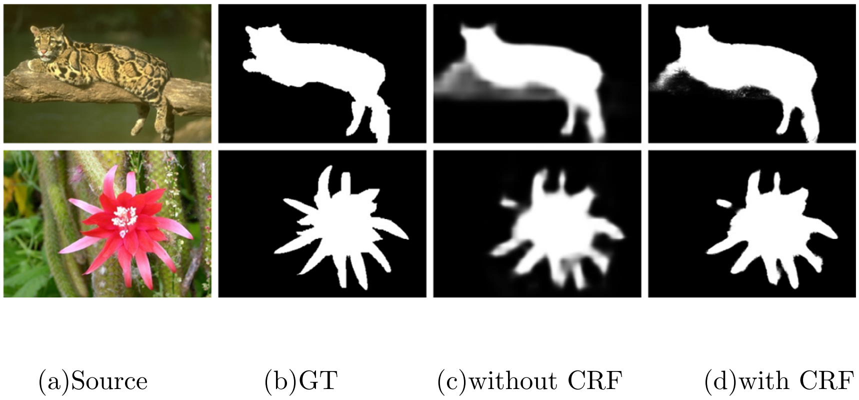

Fig. 5. Comparison of saliency detection results with and without CRF.

Table 1
Average F-measure values and average MAE values of our model and the other nine state-of-the-art saliency models on four challenging datasets. The top three results are shown in red, green and blue, respectively.

Methods	DUTS-TE		ECSSD		HKU-IS		PASCAL-S	
	F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE
Deep learning								
OURS	**0.819**	**0.065**	**0.926**	**0.045**	**0.913**	**0.038**	**0.833**	**0.101**
Amulet	0.740	0.085	**0.891**	**0.059**	**0.874**	**0.051**	**0.797**	**0.105**
DCL	**0.745**	**0.082**	0.862	0.075	0.859	0.064	0.774	0.121
DGRL	**0.803**	**0.050**	**0.906**	**0.041**	**0.890**	**0.036**	**0.818**	**0.082**
DS	0.731	0.090	0.807	0.122	0.838	0.080	0.678	0.179
MCDL	0.582	0.105	0.756	0.106	0.716	0.099	0.650	0.156
RFCN	0.734	0.087	0.850	0.901	0.850	0.074	0.771	0.122
UCF	0.718	0.112	0.877	0.069	0.858	0.062	0.778	0.119
cGANw	0.464	0.182	0.750	0.113	0.708	0.109	0.383	0.293
Non-deep learning								
DRFI	0.589	0.155	0.752	0.164	0.728	0.145	0.630	0.209

can be obtained by combining the precision and recall with β^2 as the balance factor, which is defined as:

$$F_\beta = \frac{(1+\beta^2)Precision \times Recall}{\beta^2 Precision + Recall}, \quad (6)$$

where β^2 is set to 0.3 for highlighting more on precision over recall as suggested in [52]. In order to compute MAE, given a saliency map S and the corresponding ground truth G, the score of MAE can be calculated via the element-wise difference between S and G as follows:

$$MAE = \frac{1}{W \times H}\sum_{i=1}^{W}|S(i,j) - G(i,j)|, \quad (7)$$

4.3. Performance comparison

To evaluate the performance of our model, we perform qualitative and quantitative comparisons against the state-of-the-art saliency models including Amulet [19], DCL [17], DGRL [18], DS [20], DRFI [21], MCDL [16], RFCN [22], UCF [23], cGANw [27], on the ECSSD, HKU-IS, PASCAL-S datasets and the test set of DUTS dataset.

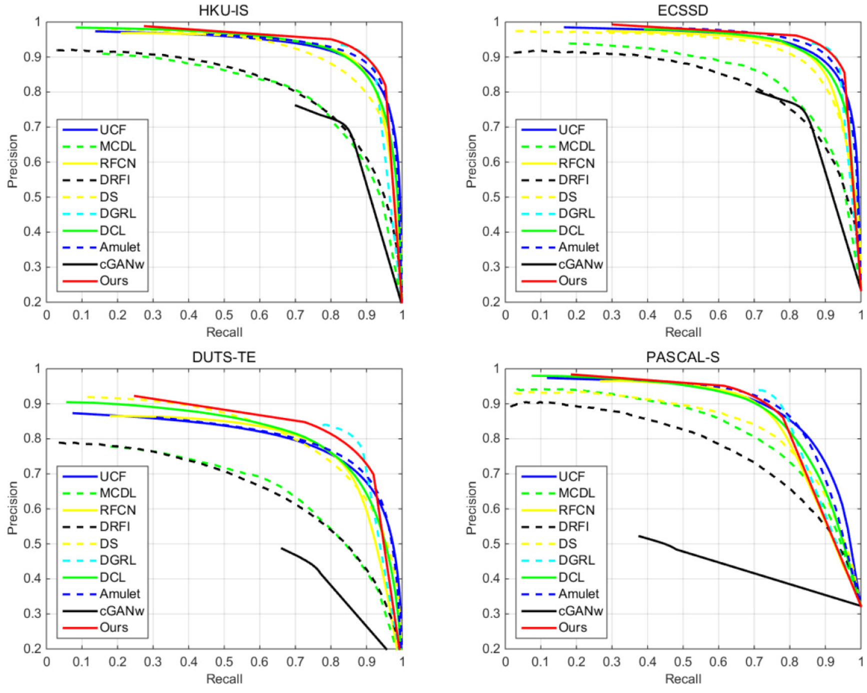

Fig. 6. PR curves of different saliency models on four challenging datasets.

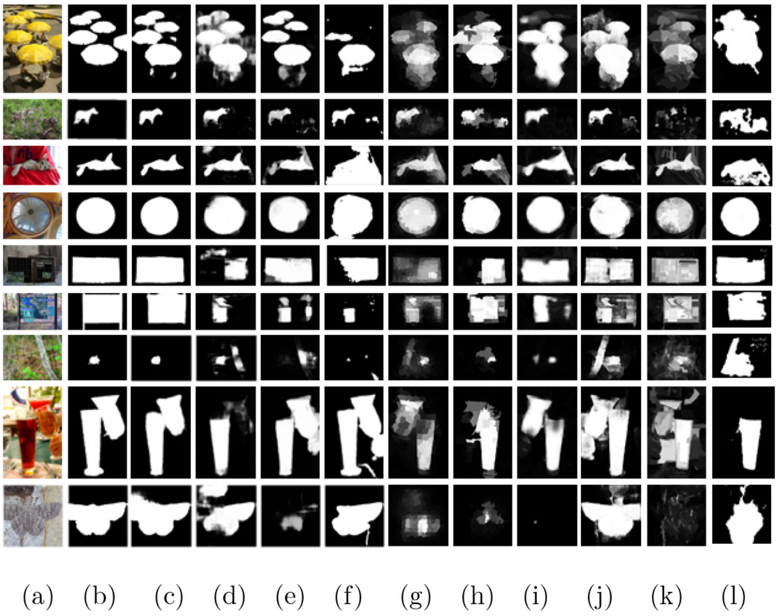

Fig. 7. Visual comparison of saliency maps on the ECSSD dataset. (a) Images, (b) ground truths, and saliency maps generated using (c) our model, (d) Amulet, (e) DCL, (f) DGRL, (g) DS (h) MCDL (i) RFCN, (j) UCF, (k) DRFI, and (l) cGANw, respectively.

Quantitative evaluation. Table 1 shows the average F-measure values and average MAE values for the aforementioned saliency models, and Fig. 6 presents their PR curves. According Table 1 and Fig. 6, we can find that our model achieves the favorable performance against the state-of-the-art saliency models. In terms of F-measure, compared to all the other saliency models, our model improves the average F-measure with a considerable margin on all the four datasets. In terms of MAE, our model takes the 2^{nd} place on all the four datasets, while slightly inferior to the DGRL model. In terms of PR curve, our model performs the best on the HKU-IS dataset and the ECSSD dataset, while on the other two datasets, DGRL performs better than our model. Considering the overall performance on all the four datasets with all the three evaluation criteria, we can conclude that our model and DGRL achieve a similar saliency detection performance, and outperform all the other state-of-the-art saliency models.

Visual comparison. To visually compare saliency maps generated using our model and the aforementioned state-of-the-art saliency models, we present some examples including salient objects with different sizes (row 1), complicated scenes (row 2 and row 8), low contrasts between objects and background (row 9) and so on, in Fig. 7. More examples including multiple salient objects are shown in Fig. 8. According to the visual comparison results, we can observe that our model can generate more accurate saliency maps for many complicated images.

4.4. Timing

Most of the existing deep learning based saliency models are time-consuming, and our model performs efficiently during both training and testing. For training our model, it only takes about 0.55 h when the model becomes convergence as shown in Table 2.

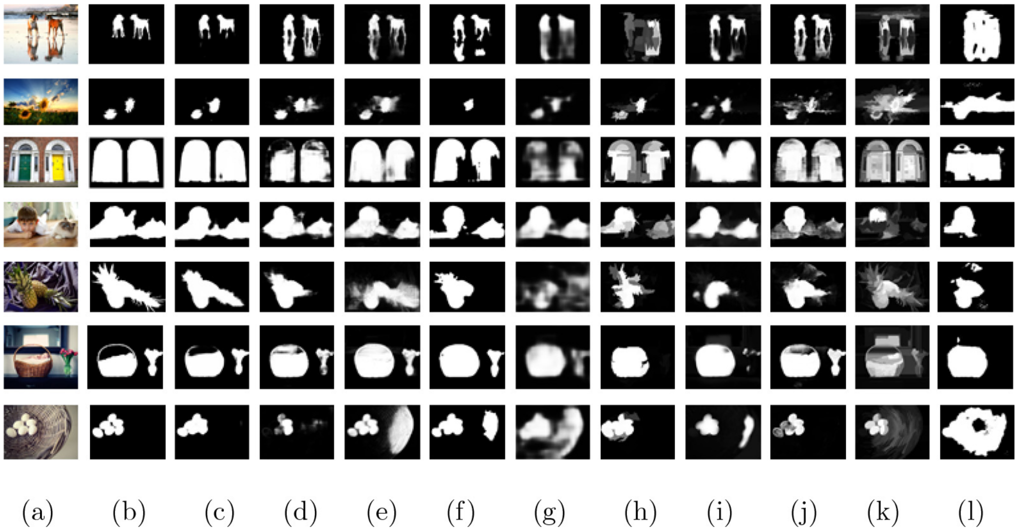

Fig. 8. Visual comparison of saliency maps on the HKU-IS dataset. (a) Images, (b) ground truths, and saliency maps generated using (c) our model, (d) Amulet, (e) DCL, (f) DGRL, (g) DS, (h) MCDL (i) RFCN, (j) UCF (k) DRFI, and (l) cGANw, respectively.

Table 2
Hardware configuration, training time and testing time of each model.

Model	DCL	DS	Amulet	MCDL	UCF	DRFI	Our
Hardware	Titan Black	Tesla K40C	GTX 1080	Titan X	Titan X	Core i5	Titan Xp
Training time (h)	~ 25	~ 72	~ 16	~ 31	~ 23	~ 24	~ 0.55
Testing time (s)	~ 1.5	~ 0.356	~ 0.063	~ 1.6	~ 0.143	~ 10	~ 0.258

Table 3
Ablation study in terms of average F-measure and average MAE.

Dataset	Metric	Our model w/o D	Our model w/o CRF	Our model
PASCAL-S	F_β	0.824	0.826	0.833
	MAE	0.104	0.110	0.101
DUT-OMRON	F_β	0.766	0.779	0.787
	MAE	0.085	0.080	0.074
HKU-IS	F_β	0.909	0.895	0.913
	MAE	0.041	0.045	0.038
ECSSD	F_β	0.915	0.914	0.926
	MAE	0.050	0.050	0.045
DUTS-TE	F_β	0.811	0.808	0.819
	MAE	0.070	0.071	0.065

Compared with the other models such as DCL, DS, Amulet, MCDL UCF, and DRFI, our model is the most efficient saliency model for training as far as we know. For testing, our model takes about 0.258 s to process an input image with a resolution of 300 × 300 on a PC with a Titan Xp GPU. Our model is slightly slower than Amulet and UCF, while faster than other models.

4.5. Ablation study

To evaluate the contribution of discriminator in our model, we perform the ablation study using the average F-measure values and average MAE values as shown in Table 3. Concretely, we obtain the saliency map using the generator, which is trained without discriminator, and the corresponding results are shown in the 3rd column (Our model w/o D) of Table 3. The results obtained using our model, in which the generator is trained with the discriminator, are shown in the rightmost column (Our model) of Table 3. Further, we also provide the results of our model without CRF, as shown in the 4th column (Our model w/o CRF) of Table 3. We can

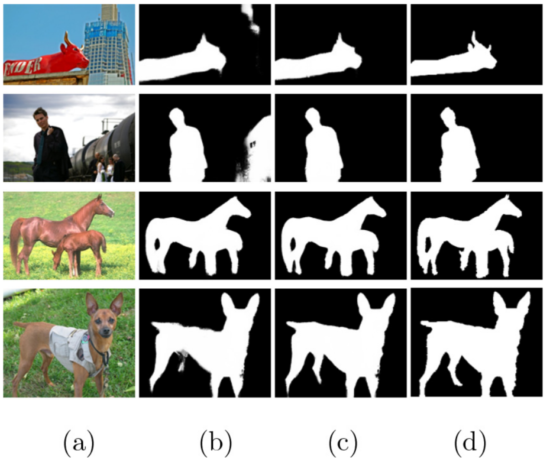

Fig. 9. Several examples for ablation study. (a) Input images, (b) saliency maps generated using the generator without discriminator, (c) saliency maps generated using our model, and (d) ground truths.

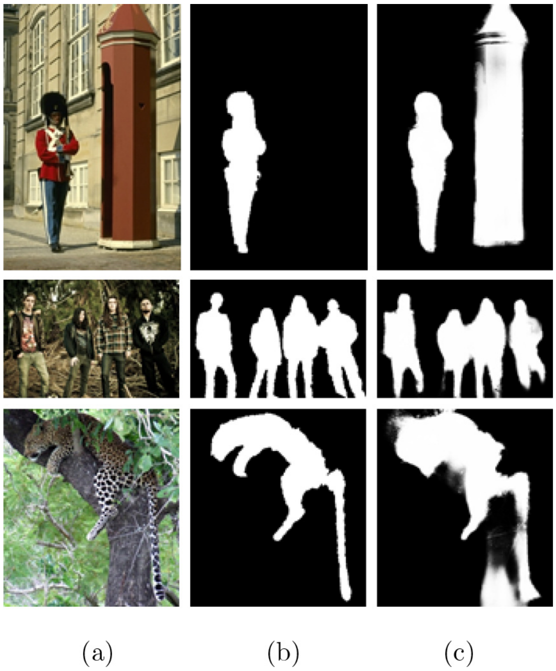

Fig. 10. Failure examples. (a) Input images, (b) ground truths, and (c) saliency maps generated using our model.

observe that our model consistently outperforms the variants, and this clearly demonstrates the effectiveness of our adversarial learning networks and the contribution of CRF. Besides, we also show several qualitative comparison results in Fig. 9. It can be seen from Fig. 9 that the saliency maps generated using the variant without discriminator falsely highlight some irrelevant regions. In contrast, the saliency maps generated using our model show the better visual quality.

4.6. Failure examples

Although our model performs well in most cases, it still fails on some challenging examples, as shown in Fig. 10. In the first example, the background region, *i.e.* the brown pillar, which is highlighted in our saliency map, is actually visually salient with distinctive colors compared to the main background. In the second and the third example, the complicated background and the low contrasts between salient objects and background result in the imprecise boundaries of salient objects in the saliency maps. For such challenging examples, our model can generally highlight the salient objects, but it is difficult for our model to obtain accurate saliency maps with well-defined boundaries.

5. Conclusion

In this paper, we propose an adversarial learning based saliency model, in which the generator is used to produce saliency map and the discriminator is applied to promote the generator. The proposed model is trained in three steps including the training of generator, the training of discriminator, and the training throughout the overall network. The key point of training process lies in the discriminator, which is designed to provide the feedback cues for refining the saliency map and accelerating the generator. Extensive experiments on the four public datasets validate the effectiveness of the proposed model.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grants 61771301 and 61901145.

References

[1] S. Hong, T. You, S. Kwak, B. Han, Online tracking by learning discriminative saliency map with convolutional neural network, in: Proc. ICML, Postech, Pohang, Korea, 2015, pp. 597–606.

[2] V. Mahadevan, N. Vasconcelos, Biologically inspired object tracking using center-surround saliency mechanisms, IEEE Trans. Pattern Anal. Mach. Intell. 35 (2013) 541–554.

[3] H. Fang, S. Gupta, F. Iandola, R.K. Srivastava., From captions to visual concepts and back, in: Proc. IEEE CVPR, Boston, 2015, pp. 1473–1482.

[4] M. Donoser, M. Urschler, M. Hirzer, H. Bischof, Saliency driven total variation segmentation, in: Proc. IEEE ICCV, Kyoto, Japan, 2009, pp. 817–824.

[5] G. Li, Z. Liu, R. Shi, W. Wei., Constrained fixation point based segmentation via deep neural network, Neurocomputing, doi:10.1016/j.neucom.2019.08.051.

[6] Z. Liu, R. Shi, L. Shen, Y. Xue, K.N. Ngan, Z. Zhang, Unsupervised salient object segmentation based on kernel density estimation and two-phase graph cut, IEEE Trans. Multimedia 14 (2012) 1275–1289.

[7] Z. Ren, S. Gao, L.-T. Chia, I.W.-H. Tsang, Region-based saliency detection and its application in object recognition, IEEE Trans. Circuits Syst. Video Technol. 24 (2014) 769–779.

[8] M.-M. Cheng, F.-L. Zhang, N.J. Mitra, X. Huang, S.-M. Hu, Repfinder: finding approximately repeated scene elements for image editing, ACM Trans. Graphics 29 (83) (2010) 1–8.

[9] X. Zhou, Z. Liu, K. Li, G. Sun, Video saliency via bagging-based prediction and spatiotemporal propagation, J. Vis. Commun. Image Represent. 51 (2018) 131–143.

[10] X. Zhou, Z. Liu, C. Gong, W. Liu, Improving video saliency detection via localized estimation and spatiotemporal refinement, IEEE Trans. Multimedia 20 (2018) 2993–3007.

[11] Z. Liu, J. Li, L. Ye, G. Sun, L. Shen, Saliency detection for unconstrained videos using superpixel-level graph and spatiotemporal propagation, IEEE Trans. Circuits Syst. Video Technol. 27 (2016) 2527–2542.

[12] Y. Wei, X. Liang, Y. Chen, X. Shen, M.-M. Cheng, Y. Zhao, S. Yan, Stc: a simple to complex framework for weakly-supervised semantic segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 39 (2017) 2314–2520.

[13] Y. Wei, H. Xiao, H. Shi, Z. Jie, J. Feng, T.S. Huang., Revisiting dilated convolution: A simple segmentation, in: Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 7268–7277.

[14] Y. Wei, J. Feng, X. Liang, M.-M. Cheng, Y. Zhao, S. Yan., Object region mining with adversarial erasing: A simple classification to semantic segmentation approach, in: Proc. IEEE CVPR, Honolulu, HI, USA, 2017, pp. 6488–6496.

[15] M. Huang, Z. Liu, L. Ye, X. Zhou, Y. Wang., Saliency detection via multi-level integration and multi-scale fusion neural networks, Neurocomputing, doi: 10.1016/j.neucom.2019.07.054.

[16] R. Zhao, W. Ouyang, H. Li, X. Wang., Saliency detection by muti-context deep learning, in: Proc. IEEE CVPR, Boston, MA, USA, 2015, pp. 1265–1274.

[17] G. Li, Y. Yu., Deep contrast learning for salient object detection, in: Proc. IEEE CVPR, Las Vegas, NV, USA, 2016, pp. 478–487.

[18] T. Wang, L. Zhang, S. Wang, H. Lu, G. Yang, X. Ruan, A. Borji., Detect globally, refine locally: A novel approach to saliency detection, in: Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 3127–3135.

[19] P. Zhang, D. Wang, H. Lu, H. Wang, X. Ruan., Amulet: Aggregating multi-level convolutional features for salient object detection, in: Proc. IEEE ICCV, Venice, Italy, 2017, pp. 202–211.

[20] Deepsaliency: Multi-task deep neural network model for salient object detection, IEEE Trans. Image Process 25 (2016) 3919–3930.

[21] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, S. Li., Salient object detection: A discriminative regional feature integration approach, in: Proc. IEEE CVPR, Portland, OR, USA, 2013, pp. 2083–2090.

[22] L. Wang, L. Wang, H. Lu, P. Zhang, X. Ruan, Saliency detection with recurrent fully convolutional networks, in: Proc. ECCV, Springer, Amsterdam, The Netherlands, 2016, pp. 825–841.

[23] P. Zhang, D. Wang, H. Lu, H. Wang, B. Yin., Learning uncertain convolutional features for accurate saliency detection, in: Proc. IEEE ICCV, Venice, Italy, 2017, pp. 212–221.

[24] J. Kuen, Z. Wang, G. Wang., Recurrent attentional networks for saliency detection, in: Proc. IEEE CVPR, Las Vegas, NV, USA, 2016, pp. 3668–3677.

[25] P. Luc, C. Couprie, S. Chintala, J. Verbeek, Semantiv segmentation using adversarial networks, in: Proc. NIPS, Barcelona, Spain, 2016.

[26] N. Souly, C. Spampinato, M. Shah., Semi and weakly supervised semantic segmentation using generative adversarial network, in: Proc. IEEE CVPR, Honolulu, HI, USA, 2017, pp. 5689–5697.

[27] Y. Ji, H. Zhang, Q.J. Wu, Saliency detection via conditional adversarial image-to-image network, Neurocomputing 316 (2018) 357–368.

[28] Q. Fan, C. Qi, Two-stage salient region detection by exploiting multiple priors, J. Vis. Commun. Image Represent. 25 (2014) 1823–1834.

[29] L. Xu, L. Zeng, H. Duan, An effective vector model for global-contrast-based saliency detection, J. Vis. Commun. Image Represent. 30 (2015) 64–74.

[30] B. Zou, Q. Liu, Z. Chen, H. Fu, C. Zhu, Surroundedness based multiscale saliency detection, J. Vis. Commun. Image Represent. 33 (2015) 378–388.

[31] L. Xu, H. Li, L. Zeng, K.N. Ngan, Saliency detection using joint spatial-color constraint and multi-scale segmentation, J. Vis. Commun. Image Represent. 24 (2013) 465–467.

[32] S. He, R. Lau, W. Liu, Z. Huang, Q. Yang, Supercnn: a super pixelwise convolutional neural network for salient object detection, Int. J. Comput. Vision 115 (2015) 330–344.

[33] G. Li, Y. Yu., Visual saliency based on multiscale deep features, in: Proc. IEEE CVPR, Boston, MA, USA, 2015, pp. 5455–5463.

[34] J. Long, E. Shelhamer, T. Darrell., Fully convolutional networks for semantic segmentation, in: Proc. IEEE CVPR Boston, MA, USA, 2015, pp. 3431–3440.

[35] Q. Zhang, J. Lin, J. Zhuge, W. Yuan, Multi-level and multi-scale deep saliency network for salient object detection, J. Vis. Commun. Image Represent. 59 (2019) 415–424.

[36] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, P.H.S. Torr, Deeply supervised salient object detection with short connections, IEEE Trans. Pattern Anal. Mach. Intell. 41 (2019) 428–815.

[37] I. Goodfellow, J. Puget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio., Generative adversarial nets, in: Proc. NIPS, Montreal, Canada, 2014, pp. 2672–2680.

[38] C. Ledig, L. Theis, F. Huszar., Photo-realistic single image super-resolution using g generative adversarial network, in: Proc. IEEE CVPR, Honolulu, HI, USA, 2017, pp. 105–114.

[39] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, T.S. Huang., Generative image inpainting with contextual attention, in: Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 5505–5514.

[40] B. Dolhansky, C.C. Ferrer., Eye in-painting with exemplar generative adversarial networks, in: Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 7902–7911.

[41] Y. Li, S. Liu, J. Yang, M.-H. Yang., Generative face completion, in: Proc. IEEE CVPR, Honolulu, HI, USA, 2017, pp. 5892–5900.

[42] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, B. Catanzaro., High-resolution image synthesis and semantic manipulation with conditional gans, in: Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 8798–8807.

[43] T. Xiao, J. Hong, J. Ma, Elegant: exchanging latent encoding with gan for transferring multiple face attributes, in: Proc. ECCV, Springer, Munich, Germany, 2018, pp. 172–187.

[44] S. Xie, R. Girshick, P. Dollar, Z. Tu, K. He., Aggregated residual transformations for deep neural network, in: Proc. IEEE CVPR, Honolulu, HI, USA, 2017, pp. 5897–5995.

[45] M.-M. Cheng, N.J. Miitra, X. Huang, P.H.S. Torr, S.-M. Hu, Global contrast based salient region detection, IEEE Trans. Pattern Anal. Mach. Intell. 37 (2015) 569–582.

[46] P. Krahenbuhl, V. Koltun., Efficient inference in fully connected crfs with gaussian edge potentials, in: Proc. NIPS, Granada, Spain, 2011, pp. 109–117.

[47] L.C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully conntected crfs, IEEE Trans. Pattern Anal. Mach. Intell. 40 (2018) 834–848.

[48] Q. Yan, L. Xu, J. Shi, J. Jia., Hierarchical saliency detection, in: Proc. IEEE CVPR, Portland, OR, USA, 2013, pp. 1155–1162.

[49] Y. Li, X. Hou, C. Koch, J.M. Rehg, A.L. Yuille., The secrets of salient object segmentation, in: Proc. IEEE CVPR, Columbus, OH, USA, 2014, pp. 280–287.

[50] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, X. Ruan., Learning to detect salient objects with image-level supervision, in: Proc. IEEE CVPR, Honolulu, HI, USA, 2017, pp. 136–145.

[51] N. Otsu, A threshold selection method from gray-level histograms, IEEE Trans. Syst. Man Cybern. 9 (1979) 62–66.

[52] R. Achanta, S. Hemami, F. Estrada, S. Susstrunk., Frequency-tuned salient region detection, in: Proc. IEEE CVPR, Miami, FL, USA, 2009, pp. 1597–1604.