Differential-Evolution-Based Generative Adversarial Networks for Edge Detection

Wenbo Zheng [1,3], Chao Gou [2], Lan Yan [3,4], Fei-Yue Wang [3,4]
[1] School of Software Engineering, Xi'an Jiaotong University
[2] School of Intelligent Systems Engineering, Sun Yat-sen University
[3] The State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences
[4] School of Artificial Intelligence, University of Chinese Academy of Sciences
zwb2017@stu.xjtu.edu.cn;gouchao@mail.sysu.edu.cn;yanlan2017@ia.ac.cn;feiyue.wang@ia.ac.cn

Abstract

Since objects in natural scenarios possess various scales and aspect ratios, learning the rich edge information is very critical for vision-based tasks. Conventional generative adversarial networks (GANs) based methods for edge detection don't perform so well due to model collapse. In order to capture rich edge information and avoid model collapse as much as possible, we consider the learning of GANs as an evolutionary optimization and propose a novel method termed as differential-evolution-based generative adversarial networks (DEGAN) for richer edge detection. In particular, built upon GANs structure, we introduce an improved differential evolutionary algorithm to refine the input of generator, with fitness function evaluated by the discriminator. Experimental results on the well-known BSDS500 and NYUD benchmarks indicate our proposed DEGAN can achieve state-of-the-art performance while retaining a fast speed and validate its simplicity, effectiveness, and efficiency. The high quality of our results on edge detection with proposed DEGAN may promise to make other vision-based tasks work better.

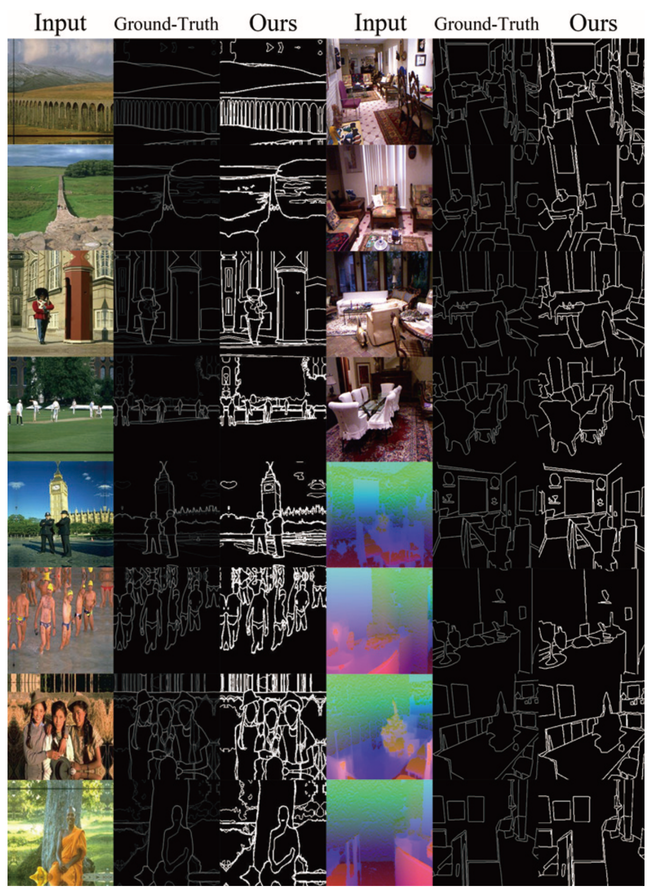


Figure 1. Some results of our model. The first three columns show example test results in the BSD500 dataset; the next three columns show example test results in the NYUD dataset.

1. Introduction

Edge detection, which aims to extract visually salient edges and object boundaries from natural images, has remained as one of the main challenges in computer vision for several decades [1]. Edge detection is one of the most important steps in vision systems. Its importance arises from the fact that edge often indicates the physical extent of the object within the image. It is considered a low-level technique, but this step determines the accuracy of the tasks of image processing, image analysis, and pattern recognition.

*This work is supported by National Natural Science Foundation of China (61806198, 61533019, U1811463). *(Corresponding author: Chao Gou).*

The history of computational edge detection is extremely rich. We now highlight a few representative works that have proven to be of great practical importance. These existing works fall into three categories. The first kind is the early pioneering methods, such as the widely Canny detector [6], zero-crossing [34], and the Sobel detector [24]. The second kind of methods is driven by information theory, such as the Statistical Edges [25], Pb [32], and gPb [1]. Third kind are based on machine learning, such as the BEL [11], Multi-scale [37], Sketch Tokens [27], and Structured Edges [13]. Also, there has been a recent wave of development using convolutional neural networks (CNN) that emphasize the importance of automatic hierarchical feature learning, including N^4-Fields [15], Deep-Contour [41], DeepEdge [4], and CSCNN [22].

The first kind of methods mainly focuses on the utilization of low-level representative features such as intensity and color gradients [29, 53]. These methods always do not perform so well and are difficult to satisfy today's applications, such as salient object detection [48] and mobile vision [26]. The second type of methods mainly focus on features arrived at through careful manual designed using information theory. However, it is developed based on handcrafted features, which has limited ability to represent high-level information for semantically meaningful edge detection. The third one can overcome this problem and principally uses CNN to get edge detection. But this learning algorithm is that the final training result depends on the initial inputs and thresholds to a large extent.

In 2014, Goodfellow et al. introduced a powerful class of generative models named generative adversarial networks (GANs) [16,46]. Over the years, generative adversarial networks have been successfully applied in the computer vision field [46,51]. Generative adversarial networks are powerful in image and vision computing [9, 20, 44, 47]. However, GANs are looking for Nash equilibrium, and there is no theoretical proof that balance and convergence can always be achieved [30]. Hence, there are two main challenges for edge detection using GANs:

(1) The training result easily falls into the local minimum point rather than into the global optimum [23].

(2) GANs may fall into mode collapse [33].

Evolutionary computation is a kind of methods for global optimization inspired by biological evolution. To address the aforementioned challenges in edge detection using GANs, we explore the use of evolutionary computation (differential evolution algorithm in particular) for optimization of GANs. In this work, by regarding adversarial training procedure in GANs as an evolutionary problem, we propose to incorporate evolutionary computation in GANs architecture for efficient and effective edge detection. Some qualitative results produced by our proposed method are shown in Figure 1. The key idea is to build a generative adversarial network based on differential evolution and treat the adversarial training process as an evolutionary problem. According to the differential evolution algorithm theory [43], the discriminator acts as an environment with a fitness function (i.e., provides an adaptive loss function), and the generator evolves as the environment changes. In each adversarial (or evolutionary) iteration, the discriminator is still trained to recognize both real and false samples. However, in our approach, as parents, the generator undergoes different mutations to produce offspring to adapt to the environment. Different objective functions are intended to minimize the different distances between the generated distribution and the data distribution, resulting in different mutations. At the same time, in view of the current optimal discriminator, we measure the quality and diversity of the samples produced by the updated offspring. Finally, according to the principle of "survival of the fittest", the poorly performing offspring are removed, and the remaining well-behaved offspring (i.e., the generator's best weights) are saved and used for further training. It is worth mentioning that, our algorithm is to initialize the edge image (which can be thought of as the result of initializing a generator) and use evolutionary algorithms to optimize the results of a generator network, different from EGAN [45] which is to initialize several generator networks and use evolutionary algorithms to select generators. We conduct experiments on the BSDS500 dataset [1] and NYUD dataset [36], which are two of most widely used for edge detection datasets. These experiments show that the proposed algorithm performs better than similar works.

In general, we propose to incorporate differential evolutionary methodology into the adversarial learning of GANs. It is worthy of highlighting the contributions of our work as follows:

(1) We propose an improved differential evolution algorithm to generate the best potential edges as initialization of input of generator that makes it more easily to converge to global optimization.

(2) We propose to jointly optimize the learning of differential evolution and GANs by further introducing the loss of discriminator as the fitness function of differential evolution.

(3) Our proposed DEGAN is simple but effective and efficient, and can achieve state-of-the-art results.

2. Proposed Method

The overall proposed architecture of differential-evolution-based generative adversarial networks is shown in Figure 2. In the following, we first introduce an improved differential evolution algorithm controlling the input of the generator network. Then we described our proposed DEGAN on the basis by the theory of Wasserstein generative adversarial network and introduced the differential evolution algorithm.

2.1. Differential Evolution Algorithm

The differential evolution (DE) algorithm [40] is a population-based and direct-search algorithm for optimizing the model globally. The main working steps of the DE algorithm are consistent with other evolutionary algorithms, including mutation, crossover and selection. By the DE algorithm, our algorithm has the main differences, namely selection based on the specific fitness named $fitness_{GAN}$ that loss value of GANs.

In order to provide a better input sample to the generative network and promote evolutionary computational processes based on feedback from the network, we combine the value called $fitness_{GAN}$ of the discriminative network's loss function with the fitness function of the DE algorithm. We use half-meanfit selection operator [49], which modified from meanfit selection [55], to substitute greedy selection of traditional DE to make up losing diversity in using p_{best} strategy [52]. Note that the initial value of $fitness_{GAN}$ is 0. It is described as:

if $X_i(g) \neq X_{p_{\min}}(g)$ and $|f(V_i(g))| \leqslant fitness_{GAN}$ and $cr_i \geqslant 0.5$ **then**

$$X_i(g+1) = f(V_i(g))$$

else

$$X_i(g+1) = \begin{cases} f(V_i(g)), |f(V_i(g))| < |f(X_i(g))| \\ X_i(g), else \end{cases}$$

end if

where g is the number of current iterations, $X_i(g)$ means is the i-th individual in the current population, $X_{p_{\min}}(g)$ means lower limits of the solution, $V_i(g)$ is i-th mutant individuals in the current population, cr_i is crossover rate corresponding to the individual $X_i(g)$, f is the objective function, and $|\xi|$ is the value of the matrix ξ.

As the input of the image, we use the gray processing to get gray image as the corresponding matrix to subtract adjacent columns or adjacent rows. Let $Input$ be the image matrix to be processed, x_{ij} is the gray value of the element (i, j). In this paper, matrix adjacent row subtraction is used to obtain the objective function f, which is the gray gradient matrix of the image.

$$Input = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ x_{n1} & x_{n2} & \cdots & x_{nn} \end{bmatrix}$$

$$f = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21}-x_{11} & x_{22}-x_{12} & \cdots & x_{2n}-x_{1n} \\ \cdots & \cdots & \cdots & \cdots \\ x_{n1}-x_{n-1,1} & x_{n2}-x_{n-1,2} & \cdots & x_{nn}-x_{n-1,n} \end{bmatrix} \tag{1}$$

2.2. Differential-Evolutionary-Based Generative Adversarial Networks

We propose differential-evolution-based generative adversarial networks (DEGAN) to achieve edge detection. We view the edge detection problem as the the classification problem of image pixels. It means that we consider one binary classification problem about edge and background [7]. We aim to make our result be similar to the ground-truth as far as possible. We use the Wasserstein generative adversarial network (WGAN) to build our model. This generative network minimizes the Wasserstein distance, that is, the generative network minimizes L. Considering that the first item of L has nothing to do with the generative network, it gets two losses of WGAN:

- The loss function of the generative network: $-E_{x\sim p_g}[f_w(x)]$
- The loss function of the discriminative network: $fitness_{GAN} = E_{x\sim p_g}[f_w(x)] - E_{x\sim p_{data}}[f_w(x)]$

In our each iteration process shown in Figure 2 of edge detection, we use the original image as the input of the DE process and get the best individual image by DE process firstly. Then, we use the best individual image and Gaussian noise as the input of the generator network. The generator network generates an image called the generated image which is one of the inputs of the discriminator network. Finally, we use the generated image and ground-truth as the input of the discriminator network, and the discriminator network determines whether or not one based on the bias of ground-truth and the generated image, which is also $fitness_{GAN}$. If the decision is true, the label that generated the image is one which is the result of our work, otherwise 0. The discriminator network feedback the labels to the generator network and feedback $fitness_{GAN}$ to the DE process. The DE process, the generator network, and the discriminator network update. The above process is repeated until the generator networks parameters have not converged. In order to increase the diversity of distribution of the generator network, we update the DE process once, and then generative network parameters for ten times. For training the networks, we alternatively update the discriminative network parameters ten times, then update of the generative network parameters once.

3. Experimental Results

Implementation Details We set the learning rate as 0.00005, the clipping parameter as 0.01, the number of iterations of the critic per generator iteration as 5 and the size of batch images as 64. In the test phase, the best individual image by DE process once are used as input to the generative network to generate edge detection image results and

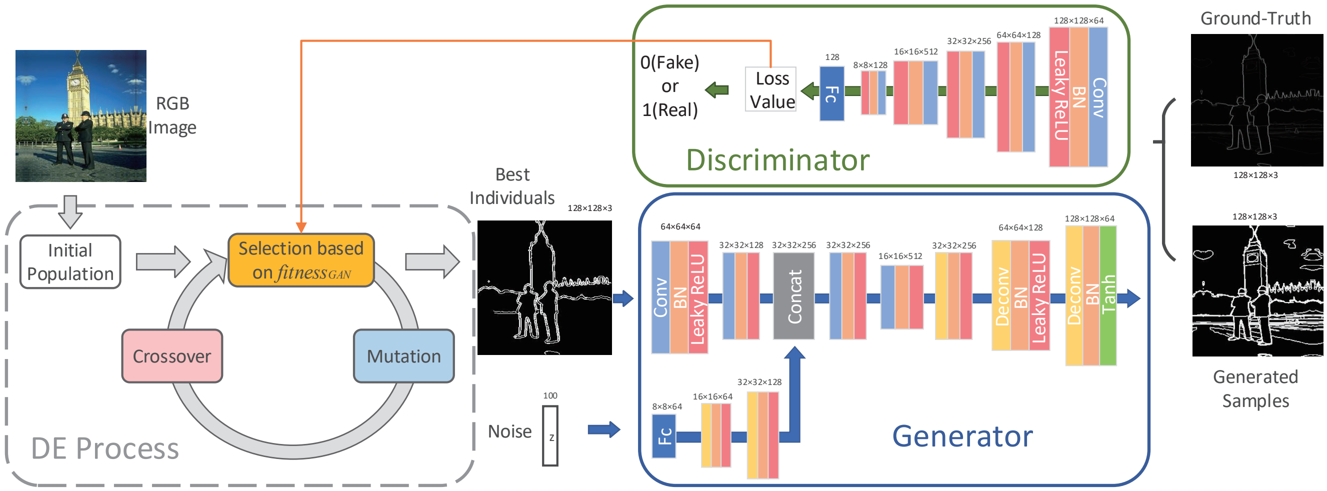


Figure 2. The architecture of differential-evolution-based generative adversarial networks

output. Besides, for the DE process, we set the number of iterations to 3000 per process. All experiments were conducted using a 4-core PC with an NVIDIA GTX 970 GPU, 16GB of RAM, and Ubuntu 16.

Quantitative Evaluation Criteria We evaluated the performance of our algorithm both on in terms of its accuracy and runtime. Edge detection accuracy is evaluated using four standard measures: fixed contour threshold (ODS), a per-image best threshold (OIS), average precision (AP), frames per second (FPS) in the test phase.

Datasets We design Ablation experiments on the BSDS500 dataset [1]. And we also design two comparison experiments on the BSDS500 dataset [1] and NYUD dataset [36], which are the large scale datasets. The information on these two databases is as follows:

- BSDS500 [1] is a widely used dataset in edge detection. It is composed of 200 training, 100 validation and 200 test images, and each image is labeled by 4 to 9 annotators. We utilize the training and validation sets for finetuning, and test set for evaluation. Data augmentation is the same as HED [53]. In particular, we rotate the images to 16 different angles and crop the largest rectangle in the rotated image. We also flip the image at each angle, leading to an augmented training set that is a factor of 32 larger than the unaugmented set. During testing, we operate on an input image at its original size. According to the work of HED [53] and RCF [29], we mix augmentation data of BSDS500 with flipped PASCAL VOC Context dataset [35] as training data.

- NYUD dataset [36] is composed of 1449 densely labeled pairs of aligned RGB and depth images. Recently many works have conducted edge evaluation on it, such as Structured Edges [13]. Gupta et al. [17] split NYUD dataset into 381 training, 414 validation and 654 testing images. We follow their settings and train our network using training and validation sets in full resolution as in HED [53] and RCF [29]. In other words, we rotate the images and corresponding annotations to 4 different angles (0, 90, 180 and 270 degrees) and flip them at each angle and we increase the maximum tolerance allowed for correct matches of edge predictions to ground truth from 0.0075 to 0.011 during evaluation.

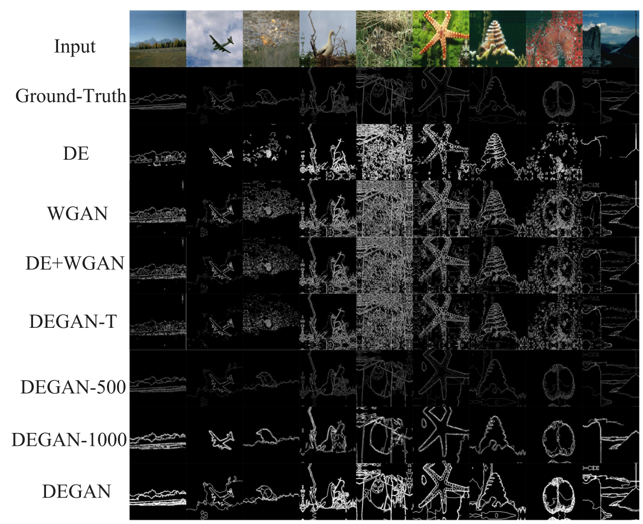


Figure 3. The Illustration of the qualitative results of the ablation study

3.1. Ablation Study

We conduct a series of ablation studies to evaluate the importance of each component in the proposed method on the BSDS500 dataset. Figure 3 is the result of our ablation

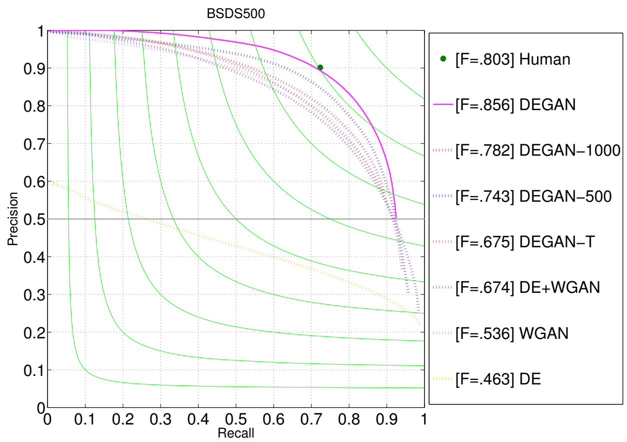


Figure 4. Precision/recall curves of Ablation study on BSDS500 dataset

Table 1. Results of Ablation study on BSDS500 dataset

	ODS	OIS	AP
DE	0.463	0.433	0.373
WGAN	0.536	0.578	0.588
DE+WGAN	0.674	0.678	0.681
DEGAN-T	0.675	0.677	0.681
DEGAN-500	0.743	0.778	0.758
DEGAN-1000	0.782	0.782	0.782
DEGAN	**0.856**	**0.873**	**0.889**

experiment. The quantitative evaluations of our ablation experiment are shown in Figure 4 and Table 1. As shown in Figure 3, Figure 4, and Table 1, it is obvious that the results of our framework are better than others.

Notation "DE" means we only use the traditional DE process to achieve edge detection. "WGAN" means we only use the WGAN to achieve edge detection. "DE+WGAN" means we use a two-stage algorithm. In the first phase, we use traditional DE process, and get the result of the first phase as the input to WGAN. Then, we train WGAN to achieve edge detection. "DEGAN-T" means we use traditional DE process to optimize WGAN to achieve edge detection. In other words, we cannot use the process of our proposed selection based $fitness_{GAN}$ in "DEGAN". " DEGAN-λ " where λ means we set the number of iterations. Note that other settings remain unchanged during the Ablation study.

Qualitative Evaluation on Ablation Study From Figure 3, we find it that the results of DEGAN are better and has a richer edge than others. For the "grassland" image (the first row), we can see that the result of the DEGAN algorithm is clear and smooth. Compare with "DE" and "WGAN", the result of "DE+WGAN" has the edges of the piles in the distant woods and adjacent grasslands. The results of "DE+WGAN" and "DEGAN-T" are similar. These show that our DE process DE has a significant optimization on effect training together with traditional DE and WGAN for edge detection. *Therefore, the design of training WGAN with DE process has a significant influence on WGAN for edge extraction.* As increasing the number of iterations, the edges of the piles in the distant woods and near grasslands is gradually clear using the DEGAN algorithm.*Therefore, the design of our DE process has a significant improvement for edge extraction.*

Quantitative Evaluation on Ablation Study To further clarify the contribution of the design of our DE process, we analyze the following two aspects: comparison of tasks whether there is our DE process and comparison of tasks where the influence of the DE process:

Comparison of Tasks Whether There Is Our DE Process Figure 4 and Table 1 show the evaluation results. The performance of the human eye in edge detection is known as 0.803 ODS F-measure. When compared with "DE" and "WGAN", ODS F-measures of "DE+WGAN" is 21.1% and 13.8% higher than it, respectively. *It is clear that DE has a significant optimization effect on WGAN for edge extraction.* When compared with "DE+WGAN", ODS F-measures of "DEGAN-T" is 0.1% higher than it. *This shows the effect of training together with traditional DE and WGAN is similar to the effect of two-stage training separately.* Compared "DEGAN-T" and "DE+WGAN", ODS F-measures of "DEGAN" is 18.3% and 18.2% higher than it, respectively. *It is obvious that our DE process DE has a significant optimization on the effect of training together with traditional DE and WGAN for edge detection. Therefore, the design of training WGAN with DE process has a significant influence on WGAN for edge extraction. In OIS and AP, we can get a similar conclusion.*

Comparison of Tasks Where the Influence of Our DE Process According to Figure 4 and Table 1, when compared with "DEGAN-500" and "DEGAN-1000", ODS F-measures of "DEGAN" is 11.3% and 7.4% higher than it, respectively. *This shows that as the number of iterations increases, the effect of our algorithm is getting better and better. Therefore, the design of our DE process has a significant improvement for edge extraction. In ODS and AP, we can get a similar conclusion.*

Stability Analysis To further analyze the stability of our algorithm, we calculated the ODS, OIS and AP values of our model for each iteration (5000 times in total) and obtained Figure 5. As shown in Figure 5, we find that the overall trend is that the values of the ODS, OIS and AP is getting larger, and the trend of the three values is generally consistent. *This shows that the performance of our algorithm is steadily improved. Therefore, our algorithm has better stability.*

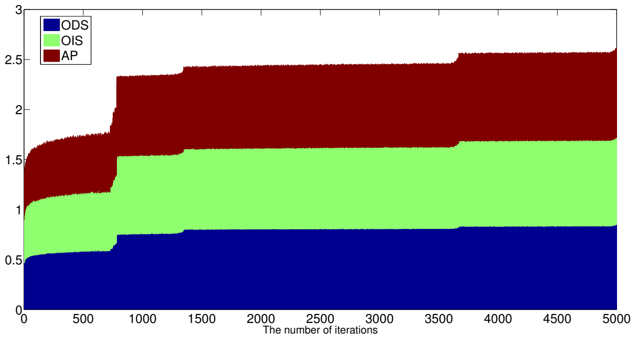


Figure 5. Stack histogram of ODS, OIS, and AP as the number of iterations increases

Figure 6. Qualitative comparison among different edge detection algorithms on BSDS500 dataset

3.2. Comparison with State-of-the-Art Methods

We design two experiments on the BSDS500 dataset [1] and NYUD dataset [36]. On the BSDS500 dataset, our algorithm compared with the some methods which are Roberts [39], Sobel [24], MShift [8], Canny [6], EGB [14], NCut [42], BEL [11], Pb [32], ISCRA [38], gPb-UCM [2] , SE [12], MCG [3], OEF [19], DeepEdge [4], DeepContour [41], HFL [5], HED [53], COB [31], RCF [29], RCF-MS [28,29], IRHED-MultiScale [50], LPCB [10], ContourGAN [54], and BDCN [21]. On NYUD dataset, our algorithm compared with the some methods which are gPb-UCM [2], OEF [19], gPb+NG [17], SE [12], SE+NG+ [18], HED [53], LPCB [10], RCF [29], ContourGAN [54], and BDCN [21].

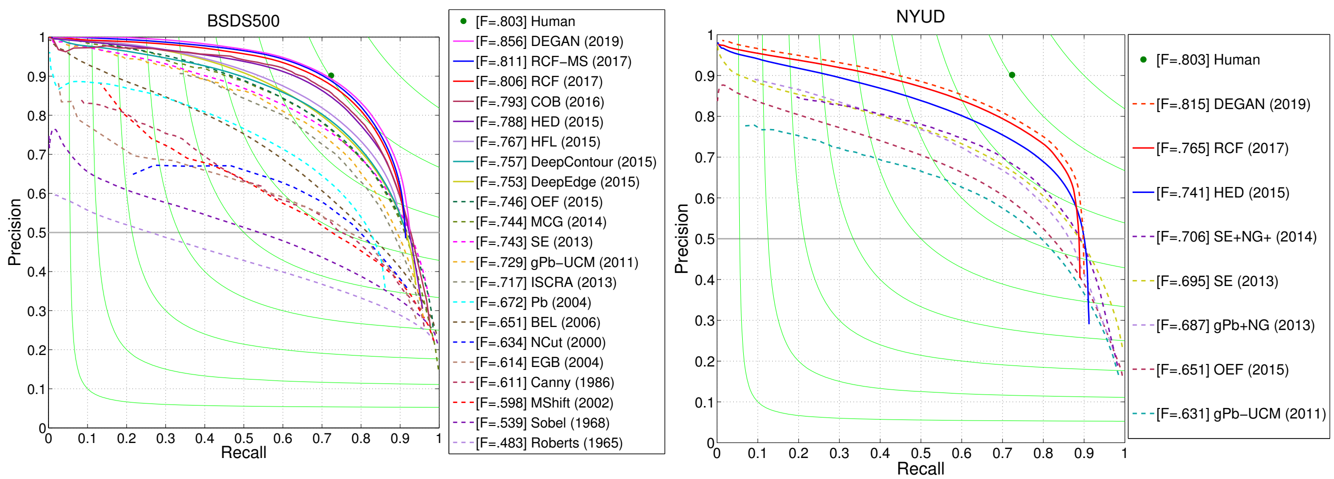
3.2.1 BSDS500 Dataset

Qualitative Comparison Evaluation on BSDS500 Dataset *From Figure 6, we find it that the results of DEGAN are better and has a richer edge than others.* For the "aircraft" image (the second row), we can see that the result of the DEGAN algorithm is clear and smooth. The result of the RCF algorithm also gets the edge of clouds, but the noise of the edge of clouds is larger. For the "starfish" image (the sixth row), we can see that the DEGAN algorithm can achieve preferable results and the edge of the reef next to the starfish and the spine, tumor or scorpion of the starfish itself is clearer than results produced by other algorithms.

Quantitative Comparison Evaluation on BSDS500 Dataset Figure 7(a) shows the evaluation results. The performance of the human eye in edge detection is known as 0.803 ODS F-measure. When compared with RCF [29] and RCF-MS [28, 29], ODS F-measures of DEGAN increase 5.0% and 4.5%, respectively. Moreover, the precision-recall curves of our methods are also higher than RCF's and RCF-MS's. *These results demonstrate the effectiveness of our richer edge features significantly. This shows that the improved method of using the DE algorithm to improve generative adversarial networks is effective and robust.*

We show statistic comparison in Table 2. In ODS F-measure, DEGAN algorithm ranks first in all algorithms. The ODS F-measure of DEGAN is 4.5%, 5.0%, 5.0% higher than RCF-MS, RCF and BDCN, respectively. In OIS and AP, we can get a similar conclusion. For FPS in the test phase, DEGAN algorithm ranks third in all algorithms and has only 20 FPS. *According to the experimental statistic results, our algorithm performs better over other algorithms, and in terms of efficiency, it meets the requirements of real-time processing.*

All in all, through the results of the experiment on BSD500 dataset, we can see that our algorithm is robust and effective. From this point, it can be proved that the way using the DE algorithm to improve the robustness of generative adversarial networks are effective. Therefore, our strategy is successful in edge extraction issues.



(a) Precision/recall curves on BSDS500 dataset

(b) Precision/recall curves on NYUD dataset

Figure 7. Precision/recall curves on two datasets

Figure 8. Qualitative comparison among different edge detection algorithms on NYUD dataset

3.2.2 NYUD Dataset

To further validate the effectiveness of our proposed DEGAN, we conduct comparisons with other state-of-the-art on NYUD dataset.

Qualitative Comparison Evaluation on NYUD Dataset *From Figure 8, we can find it that the results of DEGAN are better and has a richer edge than others.* For the "conference room" image (the second column from the left), we see that the edges of the seats and projectors generated by our DEGAN algorithm are clear, while the edges extracted by other algorithms have different noise or blur. For the "shadow wall" image (the third column from the right), we find that with the DEGAN algorithm the edges of the TV and the bookshelf are clear, while the edges extracted by other algorithms are more blurred than the DEGAN algorithm for the bookshelf. From other images, we can get a similar conclusion.

Quantitative Comparison Evaluation on NYUD Dataset Figure 7(b) shows the evaluation results. The performance of the human eye in edge detection is known as 0.803 ODS F-measure. When compared with RCF [29] and HED [53], ODS F-measures of DEGAN are 5.0% and 7.4% higher than it, respectively. Moreover, the precision-recall curves of our methods are also higher than RCF's and HED's. *These results demonstrate the effectiveness of our richer edge features significantly. This shows that the improved method of using the DE algorithm to improve generative adversarial networks is effective and robust.*

We show statistic comparison in Table 3. In ODS F-measure, DEGAN algorithm ranks first in all algorithms. The ODS F-measure of DEGAN is 5.0%, 6.7% higher than RCF and BDCN respectively. In OIS and AP, we can get a similar conclusion. For FPS in the test phase, DEGAN algorithm ranks second in all algorithms and has only 25 FPS. It means DEGAN is not better than RCF. *Therefore, according to the statistic experimental results, our algo-*

Table 2. The statistic comparison on BSDS500 dataset

Method \ Evaluation	ODS	OIS	AP	FPS
Roberts [39]	0.483	0.513	0.413	1/5
Sobel [24]	0.539	0.575	0.498	1/5
MShift [8]	0.598	0.645	0.497	1/5
Canny [6]	0.611	0.676	0.52	*28*
EGB [14]	0.614	0.658	0.564	10
NCut [42]	0.634	0.664	0.422	1/4
BEL [11]	0.651	0.674	0.701	1/3
Pb [32]	0.672	0.695	0.652	1
ISCRA [38]	0.717	0.752	0.77	1/18
gPb-UCM [2]	0.729	0.755	0.745	1/240
SE [12]	0.743	0.764	0.8	2.5
MCG [3]	0.744	0.777	0.76	1/18
OEF [19]	0.746	0.77	0.815	2/3
DeepEdge [4]	0.753	0.772	0.807	1/30
DeepContour [41]	0.757	0.776	0.79	1/1000
HFL [5]	0.767	0.788	0.795	5/6
HED [53]	0.788	0.808	0.84	1/6
COB [31]	0.793	0.819	*0.849*	**30**
RCF [29]	*0.806*	0.823	0.839	**30**
RCF-MS [28]	*0.811*	*0.83*	0.846	8
IRHED-MultiScale [50]	0.804	0.824	*0.869*	-
LPCB [10]	0.8	0.816	-	**30**
ContourGAN [54]	0.802	*0.831*	-	18
BDCN [21]	*0.806*	0.826	0.847	-
DEGAN	**0.856**	**0.873**	**0.889**	*20*

Table 3. The comparison with some competitors on NYUD dataset

Method \ Evaluation	ODS	OIS	AP	FPS
gPb-UCM [2]	0.631	0.661	0.562	1/360
OEF [19]	0.651	0.667	0.653	1/3
gPb+NG [17]	0.687	0.716	0.629	1/375
SE [12]	0.695	0.708	0.719	5
SE+NG+ [18]	0.706	0.734	0.549	1/15
HED [53]	0.741	0.757	*0.749*	*20*
RCF [29]	*0.765*	*0.78*	0.76	**30**
LPCB [10]	0.739	0.754	-	**30**
ContourGAN [54]	0.715	0.731	-	-
BDCN [21]	*0.748*	*0.763*	*0.77*	-
DEGAN	**0.815**	**0.83**	**0.81**	*25*

rithm is more excellent than other algorithms, and in terms of time, it meets the requirements of real-time processing applications.

Through the results of the experiments on two datasets, we can see that our algorithm is robust and effective. From this point, it can be proved that the way using the DE algorithm to improve the robustness and effectiveness of generative adversarial networks.

4. Conclusion

In this paper, we propose Differential-Evolution-Based Generative Adversarial Networks for richer edge detection. The key idea of proposed DEGAN is to incorporate evolutionary computation into the adversarial learning for edge detection. With the introduced differential evolution algorithm with fitness function of loss from discriminator, we can leverage the power of differential evolution to effectively avoid local optimization and employ adversarial learning with limited mode collapse to learn a better edge generator. Experimental results show that the method achieves good edge detection performance. Moreover, our proposed DEGAN makes it promising to be applied in vision tasks. In future research, we would like to apply our DEGAN to the other applications of computer vision and study how to implement our algorithm in a parallel platform for more complex calculations such as supercomputing.

References

[1] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):898–916, May 2011. 1, 2, 4, 6

[2] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 33(5):898–916, 2011. 6, 8

[3] P. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 328–335, 2014. 6, 8

[4] G. Bertasius, J. Shi, and L. Torresani. Deepedge: A multi-scale bifurcated deep network for top-down contour detection. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4380–4389, June 2015. 2, 6, 8

[5] G. Bertasius, J. Shi, and L. Torresani. High-for-low and low-for-high: Efficient boundary detection from deep object features and its applications to high-level vision. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 504–512, 2015. 6, 8

[6] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, Nov 1986. 2, 6, 8

[7] E. Chen, J. Wang, L. Qi, and W. Lv. A novel multiscale edge detection approach based on nonsubsampled contourlet transform and edge tracking. *Mathematical Problems in Engineering*, 2015:1–14, 2015. 3

[8] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 24(5):603–619, 2002. 6, 8

[9] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65, Jan 2018. 2

[10] R. Deng, C. Shen, S. Liu, H. Wang, and X. Liu. Learning to predict crisp boundaries. In *The European Conference on Computer Vision (ECCV)*, September 2018. 6, 8

[11] P. Dollar, Z. Tu, and S. Belongie. Supervised learning of edges and object boundaries. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1964–1971, June 2006. 2, 6, 8

[12] P. Dollár and C. L. Zitnick. Structured forests for fast edge detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1841–1848, 2013. 6, 8

[13] P. Dollr and C. L. Zitnick. Fast edge detection using structured forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(8):1558–1570, Aug 2015. 2, 4

[14] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *International journal of computer vision*, 59(2):167–181, 2004. 6, 8

[15] Y. Ganin and V. Lempitsky. N^4-fields: Neural network nearest neighbor fields for image transforms. In D. Cremers, I. Reid, H. Saito, and M.-H. Yang, editors, *Computer Vision – ACCV 2014*, pages 536–551, Cham, 2015. Springer International Publishing. 2

[16] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Networks. *ArXiv e-prints*, June 2014. 2

[17] S. Gupta, P. Arbelaez, and J. Malik. Perceptual organization and recognition of indoor scenes from rgb-d images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 564–571, 2013. 4, 6, 8

[18] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik. Learning rich features from rgb-d images for object detection and segmentation. In *European Conference on Computer Vision*, pages 345–360. Springer, 2014. 6, 8

[19] S. Hallman and C. C. Fowlkes. Oriented edge forests for boundary detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1732–1740, 2015. 6, 8

[20] X. Han, J. Lu, C. Zhao, S. You, and H. Li. Semisupervised and weakly supervised road detection based on generative adversarial networks. *IEEE Signal Processing Letters*, 25(4):551–555, April 2018. 2

[21] J. He, S. Zhang, M. Yang, Y. Shan, and T. Huang. Bi-directional cascade network for perceptual edge detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019 , in press. 6, 8

[22] J.-J. Hwang and T.-L. Liu. Pixel-wise deep learning for contour detection. *Computer Science*, 2015. 2

[23] E. P. Ijjina and K. M. Chalavadi. Human action recognition using genetic algorithms and convolutional neural networks. *Pattern Recognition*, 59:199 – 212, 2016. Compositional Models and Structured Learning for Visual Recognition. 2

[24] J. Kittler. On the accuracy of the sobel edge detector. *Image and Vision Computing*, 1(1):37 – 42, 1983. 2, 6, 8

[25] S. Konishi, A. L. Yuille, J. M. Coughlan, and S. C. Zhu. Statistical edge detection: learning and evaluating edge cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(1):57–74, Jan 2003. 2

[26] J. Lee, H. Tang, and J. Park. Energy efficient canny edge detector for advanced mobile vision applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(4):1037–1046, April 2018. 2

[27] J. J. Lim, C. L. Zitnick, and P. Dollr. Sketch tokens: A learned mid-level representation for contour and object detection. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3158–3165, June 2013. 2

[28] Y. Liu, M.-M. Cheng, X. Hu, J.-W. Bian, L. Zhang, X. Bai, and J. Tang. Richer convolutional features for edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, in press. 6, 8

[29] Y. Liu, X. H. Ming-Ming Cheng, K. Wang, and X. Bai. Richer convolutional features for edge detection. In *IEEE CVPR*, 2017. 2, 4, 6, 7, 8

[30] Y. Lou, Y. Hong, L. Xie, G. Shi, and K. H. Johansson. Nash equilibrium computation in subnetwork zero-sum games with switching communications. *IEEE Transactions on Automatic Control*, 61(10):2920–2935, Oct 2016. 2

[31] K.-K. Maninis, J. Pont-Tuset, P. Arbeláez, and L. Van Gool. Convolutional oriented boundaries. In *European Conference on Computer Vision*, pages 580–596. Springer, 2016. 6, 8

[32] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE transactions on pattern analysis and machine intelligence*, 26(5):530–549, 2004. 2, 6, 8

[33] L. Mescheder, S. Nowozin, and A. Geiger. Which training methods for gans do actually converge? In *International Conference on Machine Learning (ICML)*, 2018. 2

[34] E. D. Micheli, B. Caprile, P. Ottonello, and V. Torre. Localization and noise in edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(10):1106–1117, Oct 1989. 2

[35] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille. The role of context for object detection and semantic segmentation in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 4

[36] P. K. Nathan Silberman, Derek Hoiem and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 2, 4, 6

[37] X. Ren. Multi-scale improves boundary detection in natural images. In D. Forsyth, P. Torr, and A. Zisserman, editors, *Computer Vision – ECCV 2008*, pages 533–545, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg. 2

[38] Z. Ren and G. Shakhnarovich. Image segmentation by cascaded region agglomeration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2011–2018, 2013. 6, 8

[39] L. G. Roberts. *Machine perception of three-dimensional solids*. PhD thesis, Massachusetts Institute of Technology, 1963. 6, 8

[40] N. R. Sabar, J. Abawajy, and J. Yearwood. Heterogeneous cooperative co-evolution memetic differential evolution algorithm for big data optimization problems. *IEEE Transactions on Evolutionary Computation*, 21(2):315–327, April 2017. 3

[41] W. Shen, X. Wang, Y. Wang, X. Bai, and Z. Zhang. Deep-contour: A deep convolutional feature learned by positive-sharing loss for contour detection. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3982–3991, June 2015. 2, 6, 8

[42] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905, 2000. 6, 8

[43] V. A. Shim, K. C. Tan, and H. Tang. Adaptive memetic computing for evolutionary multiobjective optimization. *IEEE Transactions on Cybernetics*, 45(4):610–621, April 2015. 2

[44] W.-C. Tu, M.-Y. Liu, V. Jampani, D. Sun, S.-Y. Chien, M.-H. Yang, and J. Kautz. Learning superpixels with segmentation-aware affinity loss. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2

[45] C. Wang, C. Xu, X. Yao, and D. Tao. Evolutionary generative adversarial networks. *IEEE Transactions on Evolutionary Computation*, pages 1–1, 2019. 2

[46] K. Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng, and F. Y. Wang. Generative adversarial networks: introduction and outlook. *IEEE/CAA Journal of Automatica Sinica*, 4(4):588–598, Sept 2017. 2

[47] K. Wang, C. Gou, N. Zheng, J. M. Rehg, and F.-Y. Wang. Parallel vision for perception and understanding of complex scenes: methods, framework, and perspectives. *Artificial Intelligence Review*, 48(3):299–329, Oct 2017. 2

[48] X. Wang, H. Ma, X. Chen, and S. You. Edge preserving and multi-scale contextual neural network for salient object detection. *IEEE Transactions on Image Processing*, 27(1):121–134, Jan 2018. 2

[49] Y. Wang, Z. Cai, and Q. Zhang. Differential evolution with composite trial vector generation strategies and control parameters. *IEEE Transactions on Evolutionary Computation*, 15(1):55–66, Feb 2011. 3

[50] Y. Wang, X. Zhao, and K. Huang. Iterative residual network for structured edge detection. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 4183–4187, Oct 2018. 6, 8

[51] L. Wei, S. Zhang, W. Gao, and Q. Tian. Person transfer GAN to bridge domain gap for person re-identification. *CoRR*, abs/1711.08565, 2017. 2

[52] G. Wu, R. Mallipeddi, P. Suganthan, R. Wang, and H. Chen. Differential evolution with multi-population based ensemble of mutation strategies. *Information Sciences*, 329:329 – 345, 2016. Special issue on Discovery Science. 3

[53] S. Xie and Z. Tu. Holistically-nested edge detection. *International Journal of Computer Vision*, 125(1):3–18, Dec 2017. 2, 4, 6, 7, 8

[54] H. Yang, Y. Li, X. Yan, and F. Cao. Contourgan: Image contour detection with generative adversarial network. *Knowledge-Based Systems*, 164:21 – 28, 2019. 6, 8

[55] J. Zhang and A. C. Sanderson. Jade: Adaptive differential evolution with optional external archive. *IEEE Transactions on Evolutionary Computation*, 13(5):945–958, Oct 2009. 3