An algorithm for generating flame image data sets based on GAN

Kui Qin\textsuperscript{1,a}, Leping Bu\textsuperscript{1,b}, Yang Zhou\textsuperscript{1,e}, Zhengjun Yan\textsuperscript{1,d}, Can Wang\textsuperscript{1,e}, Teng Wang\textsuperscript{1,f}

\textsuperscript{1}Naval University of Engineering, Wuhan, Hubei, China
\textsuperscript{1}Naval University of Engineering, Wuhan, Hubei, China
\textsuperscript{1}Naval University of Engineering, Wuhan, Hubei, China
\textsuperscript{1}Naval University of Engineering, Wuhan, Hubei, China
\textsuperscript{1}Naval University of Engineering, Wuhan, Hubei, China
\textsuperscript{1}Naval University of Engineering, Wuhan, Hubei, China

\textsuperscript{1}email: 3160241095@stu.xaut.edu.cn; the corresponding author’s e-mail: 3160241095@stu.xaut.edu.cn

Abstract: In order to solve the problem of insufficient flame image data, this paper designs a flame image data sets generation algorithm based on generative adversarial network and studies the influence of different generator depth and training times on the flame generation effect. The effect of generative flame under different conditions was quantitatively evaluated by two indexes of Inception score and PSNR. The simulation results show that when the generator is 9 layers and training is 80 times, the Inception score is 2.06 and PSNR is 13.18. The effect of generative flame image is better at this time. Therefore, this generative adversarial network model can better realize the purpose of generating the flame image data sets.

1. Introduction

With the rapid development of artificial intelligence, more and more fire warning systems based on camera\textsuperscript{1,2} adopt the method of deep learning to recognize the flame image, so as to realize the fire warning. For deep learning, the size and number of data sets will greatly affect the final training effect. However, because the occurrence of fire is usually uncontrollable and the harm range is large, the flame data collection can not be obtained in the factory, warehouse, office and other places prone to fire, which limits the large-scale acquisition of flame image. Therefore, how to expand the existing flame image data integration in the form of simulation image is an urgent problem to be solved.

Traditional data sets generation methods such as rotation, translation, scaling, flipping and clipping for the original data can only improve the generalization ability of the model very finitely\textsuperscript{3}. Generative adversarial network\textsuperscript{4} provides a new way to solve the problem of data sets generation. The real flame image distribution is fitted by generative adversarial network, and the generator is used to output realistic samples. At present, GAN has achieved good results in data sets generation\textsuperscript{5,6}. In this paper, to solve the problem of flame image data sets generation, a highly realistic simulated flame image is generated by generative adversarial network. After the flame image is generated, it can be transplanted to different
scenes, instead of firing experiments in the field scene. The generated flame image data sets can be used to improve the accuracy of flame recognition.

2. Methods

2.1. Networks Structure

The generative adversarial network model is composed of generator and discriminator. The relationship between the generator and the discriminator is similar to the "zero-sum game", the sum of the "profit" and "loss" of both parties is zero. They have opposite learning goals during the training process. The learning goal of the generator is to learn the data distribution of real samples and generate images that conform to the distribution as much as possible. The role of the discriminator is to distinguish true samples from fake samples (images generated by the generator). The generative adversarial network uses the mutual confrontation training between the generator and the discriminator to gradually fit the true distribution of the target image, thereby generating a clear and realistic image.

In the training process of the generative adversarial network, the generator and the discriminator are trained separately. Generally, the discriminator is trained first. The training process of the discriminator is shown in Figure 1. Random noise z passes through the generator G and outputs the generated sample G(z). The generated sample G(z) passes through the discriminator D, and the probability that G(z) is a false sample is output. The real sample y passes through the discriminator and outputs the probability that y is a true sample. According to the sum of the two probabilities output by the discriminator, the gradient back propagation is performed to optimize the discriminator D.

![Figure 1 The training process of the discriminator](image)

The training process of the generator is shown in Figure 2. Random noise z passes through the generator G and outputs the generated sample G(z). The generated sample G(z) passes through the discriminator D, and the probability that G(z) is a true sample is output. According to the probability output by the discriminator, the gradient back propagation is performed to optimize the generator G(z).

Both the generator and the discriminator adopt a 5-layer structure, and the specific structure is shown in Figure 3.
In this paper, to study the influence of network depth on the generation effect, a 4-layer resnet block\cite{7} is added between the second and third layers of the generator. The resnet network introduces the idea of residual error into the convolutional neural network, thereby it resolves the contradiction between network depth and gradient disappearance and reduces the burden of network training to solve the problem of "stochastic gradient descent algorithm" that cannot be maximized in the deep learning network model. The single-layer resnet is shown in Figure 4.

\begin{align}
\min_G \max_D V(D, G) = \mathbb{E}_{x \sim P_{\text{data}}(x)} \left[ \log D(x) \right] + \mathbb{E}_{z \sim P_{\text{data}}(z)} \left[ \log \left( 1 - D(G(z)) \right) \right]
\end{align}

In this formula, $P_{\text{data}}(x)$ is the distribution of real data $x$; $P_{\text{data}}(z)$ is the distribution of noise $z$; $\mathbb{E}$ is the expected value of the solution.

When the discriminator is trained, the objective function of the discriminator is:
\[
\max_D V(D, G) = E_{x \sim p_{data}(x)} \left[ \log D(x) \right] + E_{z \sim p_{data}(z)} \left[ \log \left( 1 - D(G(z)) \right) \right] \\
(2)
\]

When the generator is trained, the objective function of the generator is:

\[
\min_G V(D, G) = E_z \left[ \log \left( 1 - D(G(z)) \right) \right] \\
(3)
\]

Network optimization algorithm uses Adam adaptive algorithm\(^8\).

3. Experiment and Results

3.1. Experiment

Since the background environment can be changed arbitrarily, this experiment will filter the background and only study the generation of flames. In this experiment, 20,000 flame maps were used on the pytorch0.4 deep learning framework and were trained on the NVIDIA GTX1050Ti GPU platform. The discriminator was set to 5 layers and the generator was 5 and 9 layers respectively. They were trained 200 times. The main parameter settings are shown in Table 1. The loss curve during training is shown in Figure 5. The results of training 0 times, 40 times, 80 times, 120 times, 160 times, and 200 times are shown in Figure 6.

| parameters        | value setting     |
|-------------------|-------------------|
| Input noise type  | Normal random noise|
| Adam learning rate| 0.0002             |
| Adam momentum     | 0.5               |
| batch_size        | 128               |

Table 1 Network parameter setting

![Figure 5 Training loss curve](image)
3.2. Results Analysis

Based on the analysis of Figure 6, it can be seen that when the model increases over time, the clarity of the generated flame image is improved, the quality is gradually improved, the distortion is gradually reduced, and the distribution between the generated flame image and the real image is also closer. However, it can be observed that the image quality of training is improved significantly from 0 to 80 times, but the image improvement is not obvious from 80 to 200 times. And as the number of network layers increases, the image quality is not obvious.

In this paper, Inception Score [9] and PSNR [10] (Peak Signal-to-Noise Ratio) are selected to evaluate the effectiveness of the generative adversarial network designed. Among them, the Inception Score is used to evaluate the diversity and clarity of the generated image. If the larger the Inception Score is, the richer the diversity and the higher the clarity of the generated image is. The PSNR is used to evaluate the quality of the generated image. If the larger the PSNR value is, the greater the distortion of the generated image is and the higher the image quality is. In this paper, when the generator has 5 layers, the model is trained 80 times and 200 times; when the generator is 9 layers, the model is trained 80 times and 200 times, each test is 200 sheets generated images. The quantitative evaluation of the results is shown in Table 2.

| generator layers | training times | IS   | PSNR |
|------------------|----------------|------|------|
| 5 layers         | 80             | 1.89 | 12.91|
|                  | 200            | 1.94 | 13.10|
| 9 layers         | 80             | 2.06 | 13.18|
|                  | 200            | 1.85 | 12.98|

It can be observed from the Table 2 that the Inception Score and PSNR value are the largest when the generator is 9-layer and training times is 80, indicating that the image definition is the highest at this time, the diversity is the best, the image quality is the best, and the distortion is the least. When the generator is 9 layers, the Inception Score and PSNR value of 200 times of training are lower than that of 80 times of training, which means that if the number of training is increased too much, the model will be over-fitted and the quality of the generated image cannot be improved. In summary, select generative adversarial network who is trained 80 times and generator for 9 layers as the flame image data sets generation algorithm. The generative flame images by this algorithm is shown in Figure 7.
4. Conclusion
Aiming at the problem of flame image data sets generation, this paper designs a data generation algorithm based on generative adversarial network to generate flame images through computer simulation. Because of the uncertainty of the model effect, this paper studies the influence of network depth and training times on the generation effect. Finally, the effectiveness of the algorithm is verified through experiments. In addition, the algorithm can also be used to generate data from other data sets, and can also be used to help solve problems such as image segmentation and target tracking.

References
[1] Wang T B L, Zhou Q, Et Al. A new fire recognition model based on the dispersion of color component[C]. IEEE International Conference on Progress in Informatics and Computing, 2015: 138-141.
[2] Xue Wu, Xiaoru Song, Song Gao, et al. Convolution neural network based on data enhancement for fire identification[J]. Science Technology and Engineering, 2020, 20(3): 1113-1117.
[3] Lecun, Bengio, Hinton. Deep Learning[J]. Nature, 2015, 521(7553): 436-444.
[4] Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets[C]. Advances in neural information processing systems, 2014: 2672-2680.
[5] He Yu, Nannan Yu. Enhancement of Chest X-ray Image Data by Using Fast Convergence GAN Based on Multi-Dimensional Convolution and Residual Unit[J]. Journal of Signal Processing, 2019, 35(12): 2045-2054.
[6] Lin Z, Zeng L, Wu Q. Image data augmentation of cervical cells based on generative adversarial networks[J]. Science Technology and Engineering, 2020, 20(28): 11672-11677.
[7] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[C]. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016: 770-778.
[8] Ruder S. An overview of gradient descent optimization algorithms[J]. arXiv preprint arXiv:1609.04747, 2016.
[9] Barratt S, Sharma R. A Note on the Inception Score[J]. ArXiv, 2018, abs/1801.01973.
[10] Huynh-Thu Q, Ghanbari M. The accuracy of PSNR in predicting video quality for different video scenes and frame rates[J]. Telecommunication Systems, 2010, 49(1): 35-48.