Journal on Artificial Intelligence JAI, vol.2, no.1, pp.1-15, 2020

Classification Algorithm Optimization Based on Triple-GAN

Kun Fang[1, 2] and Jianquan Ouyang[1, *]

Abstract: Generating an Adversarial network (GAN) has shown great development prospects in image generation and semi-supervised learning and has evolved into Triple-GAN. However, there are still two problems that need to be solved in Triple-GAN: based on the KL divergence distribution structure, gradients are easy to disappear and training instability occurs. Since Triple-GAN tags the samples manually, the manual marking workload is too large. Marked uneven and so on. This article builds on this improved Triple-GAN model (Improved Triple-GAN), which uses Random Forests to classify real samples, automate tagging of leaf nodes, and use Least Squares Generative Adversarial Networks (LSGAN) ideological structure loss function to avoid gradients disappear. Experiments were performed on the Improved Triple-GAN model and the Triple-GAN model using the MINIST, cifar10 and cifar100 datasets respectively, experiments show that the error rate of generated samples is greatly reduced. At the same time, the classification effect of the data set and the sharpness of the samples are greatly improved. And it has greatly improved the stability of model training and automation of labels.

Keywords: Triple-GAN, random forests, improved triple-GAN, chi-square distribution.

1 Introduction

Image mining is an emerging field in data mining. Image classification is the basis of data mining. Faced with a large number of image data, image classification becomes more and more important. Image classification, as one of the important means to understand image content, has been successfully applied in the fields of finance, public safety, transportation, etc, and its importance is self-evident. Faced with a large number of image data sets, it is not easy to retrieve and classify the semantic attributes of the images by manual methods, which causes problems such as inaccurate image information analysis. Image classification currently has many classification techniques, such as decision tree [Liu and Fan (2014)], minimum distance method, neural network [Alec, Luke and Soumith (2015)], fuzzy classification, support vector machine [Jitender and Julka (2017)], k-means [Amorim and Mirkin (2012)], and so on. The GAN model [Wang, Xu, Yao et al. (2019)] proposed a new height in the field of image classification, and also promoted the development of image mining technology. Based on the GAN model research, it has

[1] Key Laboratory of Intelligent Computing & Information Processing, Ministry of Education, Xiangtan University, Xiangtan, China.

[2] Hunan Meteorological Information Center, Hunan Meteorological Bureau, Changsha, China.

[*] Corresponding Author: Jianquan Ouyang. Email: kissingman1@gmail.com.

Received: 16 January 2020; Accepted: 01 April 2020.

become a research hotspot.

The generation of confrontation networks (GAN) also shows great development prospects in the field of image generation [Ma, Yu, Liang et al. (2019)] and semi-supervised learning [Leonardo and Alastair (1998)]. GAN has become an image and visual computing, speech and language processing, information security, chess competitions, especially artificial intelligence. Hot research directions in the field. Now GAN has evolved from a two-player game to a Triple-GAN game, and the Triple-GAN body includes a classifier (C), a generator (G), and a discriminator (D). Triple-GAN [Li, Xu, Zu et al. (2017)] ensures that classifiers and generators implement an optimal solution for classification from a game theory perspective and that generators can sample data in a particular class.

Triple-GAN based on the KL divergence distribution structure of the target loss function, there is still when the distribution does not intersect, the gradient disappears. Therefore, it is proposed to use the least-squares generation countermeasure network (LSGAN) to use the least-squares loss function for the discriminator. Experiments show that the LSGAN objective function can minimize the $\chi^2_{Pearson}$ distribution. Compared with conventional GAN [Zhou, Liang, Song et al. (2019)], LSGAN has two advantages: LSGAN can generate higher quality images than conventional GAN; LSGAN behaves more stably during the learning process [Mao, Li and Xie (2017)].

However, Triple-GAN still has the following problems in the application process: (1) Due to the use of the KL divergence distribution, gradients disappear easily in the case where the distribution does not intersect, which results in the termination of training when the training does not reach the desired result. Make model training unstable [Brock, Donahue and Simonyan (2019)]. (2) There is a problem that a large amount of data labels has a large workload since manually labeling a sample causes uneven label labels [Zhang, Goodfellow, Metaxas et al. (2018)].

In response to the above problems, this paper will use Random Forests [Adele, Richard Cutler and John Stevens (2011)] to optimize the Triple-GAN classification algorithm. At the same time, using LSGAN to build loss distribution function based on minimizing the chi-square distribution, we plan to carry out the following work: (1) Improve the classifier and integrate the random forest into the classifier to construct a classifier with automatic tagging. (2) Using the inference process of the LSGAN loss function, combined with the Triple-GAN construction model, it is possible to inherit the stability of the LSGAN and inherit the advantages of the Triple-GAN.

2 Improved triple-GAN model

2.1 Improved triple-GAN build

Due to the existence of the Triple-GAN model, there is a problem that when the distribution does not intersect, the gradient disappear and the manual label becomes cumbersome. Based on this, this paper will improve Triple-GAN in two aspects [Salimans, Goodfellow, Zaremba et al. (2016)]. In the first aspect, it is proposed to refer to the random forest to implement the automatic label of real samples. In the second aspect, based on the LSGAN theoretical model, the Triple-GAN objective function is optimized and improved so that

Triple-GAN inherits the LSGAN theoretical model to avoid the advantages of gradient disappearance [Sutherland, Tung, Strathmann et al. (2017)].

The Improved Triple-GAN model introduces three joint distribution networks, classifiers, discriminators, and generators [Arjovsky and Bottou (2017)]. The specific steps are as follows: (1) Randomly retrieving real samples, setting up a random forest decision tree, and automatically marking the leaves of decision trees. After the random forest decision tree is constructed, leaf node data with label and non-leaf node data with no label are input into the neural network (semi-supervised learning) for training [du Plessis, Niu and Sugiyama (2015)]. At the same time, the tagged leaf node is input to the discriminator. (2) The training sample is input into the generator so that the generator will obtain the joint distribution $P_g(x,y)$ of the tag classy and the generated data x through training. (3) $P_g(x,y)$ is input into the classifier, and the corresponding leaf node labels are found through the prediction analysis of the random forest decision tree to form the joint distribution $P_c(x,y)$ of the leaf node labels and the data generated by the classifier. (4) $P_g(x,y)$and $P_c(x,y)$are input to a discriminator and judged by a target loss function. When the target loss rate is 0.5, it indicates that the generated tags and the classified tags are all optimal for the distribution, and the training is completed at this time. Otherwise iterate the above steps and repeat the training until the training reaches the optimal training. The block diagram of the model is shown in Fig. 1.

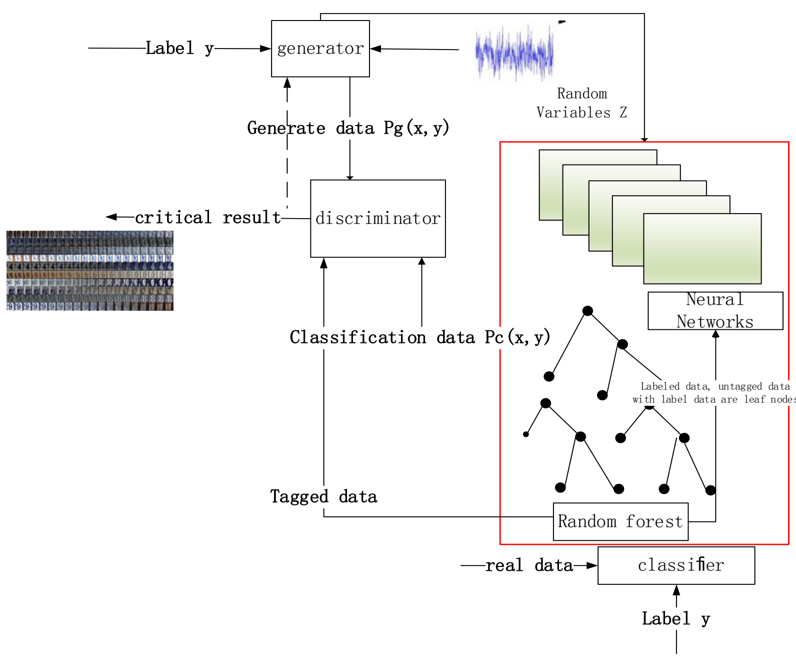


Figure 1: Improved Triple-GAN model

2.2 Improved triple-GAN model theory analysis

Using the LSGAN objective function principle, the a-b encoding scheme is used for the discriminator, where a and b are the labels of the dummy data and the real data, respectively. In the formula, C represents a classifier, G represents a generator, and D represents a discriminator (the same below). For any fixed C and G, the game's optimal discriminator D is defined by the objective function *U(C, G, D)*, as shown in Eq. (1):

$$D^{*}_{C,G}(x,y)=\frac{bP(x,y)+aP_{\alpha}(x,y)}{P(x,y)+P_{\alpha}(x,y)} \tag{1}$$

Among them $P_{a}(x,y)=(1-\alpha)P_{g}(x,y)+\alpha P_{c}(x,y)\alpha\in(0,1)$. Given classifier C and generator G, combined with the LSGAN objective function, where c denotes that G, C expect D to believe the value of the dummy data, the objective function can be rewritten as Eq. (2):

$$\begin{aligned}U(C,G,D)=&\frac{1}{2}\iint P(x,y)(D(x,y)-c)^{2}\,dydx\\&+\frac{1}{2}(1-\alpha)\iint P(y)P_{z}(z)(D(G(z,y),y)-c)^{2}\,dydz\\&+\frac{1}{2}\alpha\iint P(x)P_{C}(y\mid x)(D(x,y)-c)^{2}\,dydx\\=&\frac{1}{2}\iint P(x,y)(D(x,y)-c)^{2}\,dydx+\frac{1}{2}\iint P_{\alpha}(x,y)(D(G(x,y))-c)^{2}\,dydx\\=&f(D(x,y))\end{aligned} \tag{2}$$

When *P(x,y)=Pa(x,y)*, the global minimum function *V(C, G)* is constructed, and Eq. (2) is converted to Eq. (3):

$$\begin{aligned}2V(C,G)&\iint P(x,y)\left(\frac{(b-c)P(x,y)+(a-c)P_{\alpha}(x,y)}{P(x,y)+P_{\alpha}(x,y)}\right)^{2}dxdy\\&+\iint P_{\alpha}(x,y)\left(\frac{(b-c)P(x,y)+(a-c)P_{\alpha}(x,y)}{P(x,y)+P_{\alpha}(x,y)}\right)^{2}dxdy\end{aligned} \tag{3}$$

When b-c=1 and b-a=2, according to the derivation of LSGAN, Eq. (3) can be derived as:

$$2V(C,G)=\int_{x}\frac{\left(2P_{\alpha}(x)+\left(P(x)+P_{\alpha}(x)\right)\right)^{2}}{P(x)+P_{\alpha}(x)}=\chi^{2}_{Pearson}\left(P+P_{\alpha}\,\|\,2P_{\alpha}\right) \tag{4}$$

From Eq. (4) it can be deduced that the global minimum function V(C, G) is based on $\chi^{2}_{pearson}\left(P+P_{\alpha}\|2P_{\alpha}\right)$ distributed. Among them

$P_{a}(x,y)=(1-\alpha)P_{g}(x,y)+\alpha P_{c}(x,y)\alpha\in(0,1)$, take the integral on both sides as shown in Eq. (5):

$$\int P(x,y)=(1-\alpha)\int P_{g}(x,y)dx+\alpha\int P_{c}(x,y)dx \tag{5}$$

According to Eq. (2), the objective function pair *V(C, G)* takes the minimum value, and D makes the maximum value judgment, and the objective function is obtained as shown in Eq. (6):

$$\begin{aligned}\min_{C,G}\max_{D} U(C,G,D) = \frac{1}{2}E_{(x,y)\sim p(x,y)}\left[(D(x,y)-c)^2\right] + \frac{1}{2}\alpha E_{(x,y)\sim p_c(x,y)}\left[(D(x,y)-b)^2\right] \\ + \frac{1}{2}(1-\alpha)E_{(X,Y)\sim p_g(x,y)}\left[(D(G(y,z),y)-a)^2\right]\end{aligned} \tag{6}$$

For the Eq. (6) reasoning process description: $\boldsymbol{x}$ represents the generated data, y represents the class label, z represents the leaf node corresponding to the generated data x; $\boldsymbol{a}$ represents the generator (G) encodes the generated data, $a \in \{-1,0,1\}$, b means the classifier (C) encodes the class label, $b \in \{-1,0,1\}$; c means the generator (G) uses the objective function to determine the optimal deceptive sample data for the generated data, $c \in \{-1,0,1\}$; in this formula: $E_{(x,\,y)\sim p(x\,,y)}$ represents the joint distribution $p\ (x,y)$ of the label pair (x,y) in the discriminator (D) The expected value of D; D(x,y) represents the probability of the true sample label pair of the label pair (x,y) in the discriminator; the probability of $E_{(x,y)\sim pc\,(x,y)}$ indicates the label pair (x,y). The expected value of the joint distribution $P_c(x,y)$ in the classifier (C); $E_{(x,y)\sim pg\,(x,y)}$ means that the label pair (x,y) is in the generator (G) Expected value of the joint distribution P_g (x, y); G (x,z) represents the mapping of the label pair (y,z) in the sample space in the generator, and D (G (y,z), y) represents the label in the discriminator Probability of generated samples for (y,z) versus real samples; R_c represents the equilibrium function; α is the adjustment parameter, $\alpha \in \{0,1\}$.

When b-c=1, b-a=2, and $P(x,y)=P_a(x,\ y)$, $V(G,\ C)=0$, causing unbalanced training and making the results unoptimized. To solve this problem, construct R_c function, R_c uses $\chi^2_{Pearson}$ to arrange reliability control and distribution principle, and make discriminator and classification as the distribution equilibrium structure, namely:

$$R_c = \chi^2_{Pearson}\left(P \| P_c\right) + \sum\left(P - P_c\right)^2 \tag{7}$$

Combining the Eq. (6) and Eq. (7) to get the final objective function Eq. (8):

$$\begin{aligned}\min_{c.G}\max_{D} U(C,G,D) = \frac{1}{2}E_{(x,y)\sim p(x,y)}\left[(D(x,y)-c)^2\right] + \frac{1}{2}\alpha E_{(x,y)\sim p_c(x,y)}\left[(D(x,y)-b)^2\right] \\ + \frac{1}{2}(1-\alpha)E_{(X,Y)\sim p_g(x,y)}\left[(D(G(y,z),y)-a)^2\right] + R_C\end{aligned} \tag{8}$$

According to the theoretical analysis, the final objective function inherits Triple-GAN's triple-antagonism model structure. At the same time, it refers to the construction of the LSGAN objective function and obeys the $\chi^2_{Pearson}$ distribution.

3 Experiment analysis

3.1 Model training results

In this paper, three data sets, MINIST [MNIST Dataset (2019)], cifar10 [CIFAR-10 Dataset (2019)], and cifar100 [CIFAR-100 Dataset (2019)] were trained on the Improved Triple-GAN model and the Triple-GAN model, respectively. The following describes the training effects of different training sets:

(1) The experimental results of the Improved Triple-GAN model MINIST data set in Tab. 1 analysis: The iterative error rate of the Improved Triple-GAN is 0.02720 at the first and 0.01150 at the 100th, indicating that the error rate gradually decreases after iteration. Improved Triple-GAN discriminant loss rate, generation loss rate, classification loss rate at the first iteration, the loss rate of the three values are larger (the discriminatory loss rate: 0.2077, generated loss rate:0.9897, classification loss rate: From the 0.3316) to the 100th iteration, the difference between the three-loss rates is not significant (discrimination loss rate: 0.3463, loss generation rate: 0.3471, classification loss rate: 0.3514), indicating that the Improved Triple-GAN model is trained iteratively. The loss rate is close to balance.

Table 1: Experimental results of the Improved Triple-GAN model MINIST dataset

The Nth iteration	The Nth iteration training time (unit: S)	Distinguish loss rate	Generated loss rate	Classification loss rate	Error rate
1	190.86	0.2077	0.9897	0.3316	0.02720
10	189.23	0.2939	0.4514	0.3482	0.02300
20	188.92	0.3275	0.3794	0.3521	0.01920
50	189.00	0.3454	0.3500	0.3520	0.01670
100	189.20	0.3463	0.3471	0.3514	0.01150

The experimental results of Tab. 2 Triple-GAN model MINIST data set are analyzed: the iteration error rate of Triple-GAN is 0.05820 for the first time and 0.01360 for the 100th time, indicating that the error rate gradually decreases after iteration. At the same time, it can be seen that Triple-GAN discriminates the loss rate, the generation loss rate, and the classification loss rate. At the first iteration, the difference between the three values of the loss rate is relatively large. They are the discriminatory loss rate: 0.2021, and the generation loss rate: 0.9956. Classification loss rate: 0.3296. At the 100th iteration, there was no significant difference in the loss rates of the three categories, namely the discriminatory loss rate: 0.3462, the generated loss rate: 0.3473, and the classification loss rate: 0.3519, indicating that the Triple-GAN's loss rate through the iterative training approached equilibrium. Therefore, it can be judged that compared with Triple-GAN, the Improved Triple-GAN error rate is lower, and at the same time, it inherits the advantages of Triple-GAN balance.

Table 2: Experimental results of the Triple-GAN model MINIST dataset

The Nth iteration	The Nth iteration training time (unit: S)	Distinguish loss rate	Generated loss rate	Classificati on loss rate	Error rate
1	292.21	0.2021	0.9956	0.3296	0.05820
10	296.10	0.3029	0.4387	0.3494	0.04570
20	312.91	0.2760	0.5708	0.3371	0.04020
50	216.80	0.3459	0.3482	0.3529	0.01860
100	220.85	0.3462	0.3473	0.3519	0.01360

(2) The experimental results of Tab. 3 Improved Triple-GAN model cifar10 data set are analyzed: The iterative error rate of Improved Triple-GAN is 0.46810 for the first time and 0.57640 for the 40th time, which shows that the error rate gradually decreases after iteration. (2) Improved Triple-GAN discriminative loss rate, generation loss rate, and classification loss rate at the first iteration, the three values of the loss rate are quite different. They are the discriminatory loss rate: 0.3173, the generated loss rate: 0.4077, the classification loss rate: 0.8093. By the 40th iteration, the three values of the loss rate are not much different. They are the discriminatory loss rate: 0.3422, the generated loss rate: 0.3543, and the classification loss rate: 0.3140, which indicates that the Triple-GAN through the iterative training loss rate is close to the three. balance.

Table 3: Experimental results of the improved triple-GAN model cifar10 dataset

The Nth iteration	The Nth iteration training time (unit: S)	Distinguish loss rate	Generated loss rate	Classification loss rate	Error rate
1	2238.92	0.3173	0.4077	0.8093	0.46810
10	2240.53	0.3375	0.3688	0.7153	0.27740
20	2242.27	0.3336	0.3683	0.6853	0.23200
30	2267.38	0.3412	0.3563	0.5917	0.20330
40	2322.25	0.3422	0.3543	0.3140	0.18580

The experimental results of the Tab. 4 Triple-GAN model cifar10 data set are analyzed: the iteration error rate of Triple-GAN is 0.57640 for the first time and 0.26420 for the 40th time, indicating that the error rate gradually decreases after iteration. Triple-GAN discriminates loss rate, generation loss rate, and classification loss rate. At the first iteration, there is a large gap between the three-loss rates, which are the discriminatory loss rate: 0.3216, the generated loss rate: 0.3987, the classification loss rate: 0.8009, At the 40th iteration, the three values of the loss rate are not significantly different, they are the discriminatory loss rate: 0.3431, the generated loss rate: 0.3529, the classification loss rate: 0.3178), indicating that the loss rate of the Triple-GAN through iterative training is close to the three. balance. Therefore, compared to Triple-GAN, Improved Triple-GAN has a lower error rate, and at the same time, it inherits the balancing advantages of Triple-GAN.

Table 4: Experimental results of the Triple-GAN model cifar10 dataset

The Nth iteration	The Nth iteration training time (unit: S)	Distinguish loss rate	Generated loss rate	Classification loss rate	Error rate
1	2241.35	0.3216	0.3987	0..8009	0.57640
10	2263.57	0.3308	0.3779	0.7962	0.38620
20	2262.76	0.3365	0.3649	0.6826	0.34970
30	2271.07	0.3422	0.3550	0.7931	0.30660
40	2306.87	0.3431	0.3529	0.3178	0.26420

(3) The experimental results of Tab. 5 Improved Triple-GAN model cifar100 data set are analyzed: The iterative error rate of Improved Triple-GAN is 0.82530 for the first time and 0.35380 for the 100th iteration, indicating that the error rate gradually decreases after iteration. Improved Triple-GAN discriminant loss rate, generation loss rate, classification loss rate, the first iteration is that the loss rate of the three values are larger, they are the discriminatory loss rate: 0.383, the generated loss rate: 0.4447, the classification loss rate: 0.9969. At the 100th iteration, the loss rates of the three are not significant. They are the discriminatory loss rate: 0.3520, the generation loss rate: 0.3587, and the classification loss rate: 0.3556. This indicates that the Triple-GAN through the iterative training is nearly equal to the loss rate.

The experimental results of Tab. 6 Triple-GAN model cifar100 data set are analyzed: the iteration error rate of Triple-GAN is 0.92000 for the first time and 0.42480 for the 100th time, indicating that the error rate gradually decreases after iteration. Triple-GAN discriminates the loss rate, the generation loss rate, and the classification loss rate. In the first iteration, the difference between the three values of the loss rate is relatively large. They are the discriminatory loss rate: 0.3095, the generation loss rate: 0.4202, and the classification loss rate: 0.9998. At the 100th iteration, there was no significant difference in the loss rates of the three categories, namely the discriminatory loss rate: 0.3020, the generated loss rate: 0.3187, and the classification loss rate: 0.3256, indicating that the Triple-GAN's loss rate was nearly balanced through iterative training. Therefore, it can be judged that the Improved Triple-GAN error rate is lower than that of Triple-GAN, and Triple-GAN inherits the balancing advantages.

Table 5: Experimental results of the Improved Triple-GAN model cifar100 dataset

The Nth iteration	The Nth iteration training time (unit: S)	Distinguish loss rate	Generated loss rate	Classification loss rate	Error rate
1	2297.69	0.383	0.4447	0.9969	0.82530
20	2320.57	0.3539	0.3685	0.7512	0.66100
40	2342.24	0.351	0.3610	0.55503	0.65190
60	2371.06	0.3827	0.4199	0.3018	0.67030
80	2376.18	0.3545	0.3589	0.3508	0.42490
100	2366.07	0.3520	0.3587	0.3556	0.35380

Table 6: Experimental results of the Triple-GAN model cifar100 dataset

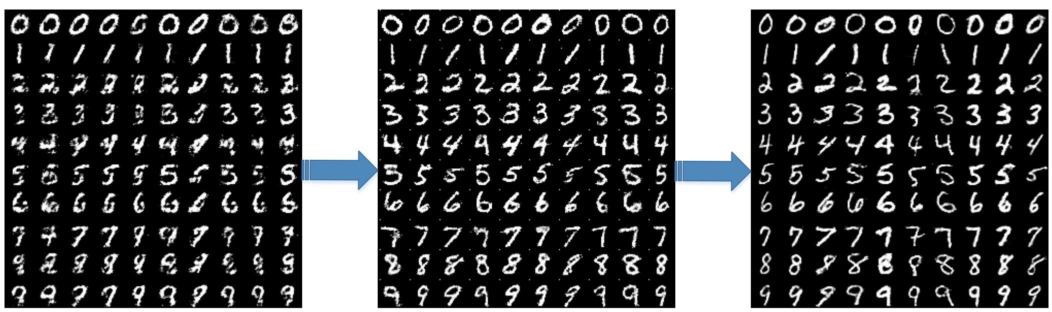
The Nth iteration	The Nth iteration training time (unit: S)	Distinguish loss rate	Generated loss rate	Classification loss rate	Error rate
1	2346.95	0.3095	0.4202	0.9998	0.92000
20	2274.75	0.3352	0.3667	0.7354	0.88170
40	2298.26	0.3195	0.3824	0.5291	0.72620
60	2898.09	0.3036	0.3288	0.3209	0.70430
80	2877.19	0.3045	0.3189	0.3108	0.52460
100	2876.07	0.3020	0.3187	0.3256	0.42480

Experiments were conducted on the Improved Triple-GAN model and the Triple-GAN model on the three data sets MINIST, cifar10, and cifar100, respectively. The experimental results show that compared with Triple-GAN, the error rate of Improved Triple-GAN is lower and inherits the balanced advantages of Triple-GAN. It can be concluded that Improved Triple-GAN can inherit the advantages of Triple-GAN and is better than Triple-GAN. Besides, through the comparison of cifar10 and cifar100 experiments, it can be seen that the more types of data, the error rate will be significantly improved. By comparing the experiment of MINIST with cifar10, we can get that the more complicated the data set, the higher the error rate.

3.2 Image effect

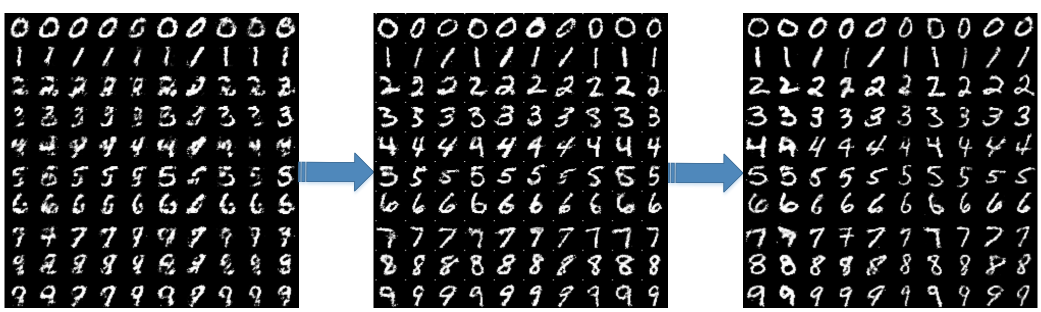
Three kinds of data sets, MINIST, cifar10, and cifar100, were analyzed on the images of the Triple-GAN model, the Triple-GAN model, and the LSGAN model. The following describes the experimental results from several aspects:

(1) Analysis of image effects trained from iterative training process, as shown in Fig. 2 Improved Triple-GAN model MINIST iterative training image effect image sharpness analysis, Improved Triple-GAN model training MINIST data set from the iterative 10 times image output is more blurred After 200 iterations, the image output is relatively clear, indicating that after iterative training, the image becomes clearer and clearer. As shown in Fig. 3 Triple-GAN model MINIST iterative training image effect image sharpness analysis, Triple-GAN model training MINIST data set from the iterative 10 times image output is relatively fuzzy to 200 iterations after the image output is relatively clear, indicating that after iterative training, the image is getting clearer. However, Improved Triple-GAN and Triple-GAN have significantly better images when they are trained for the same number of iterations.



(a) The 10th iteration (b) The 100th iteration (c) The 200th iteration

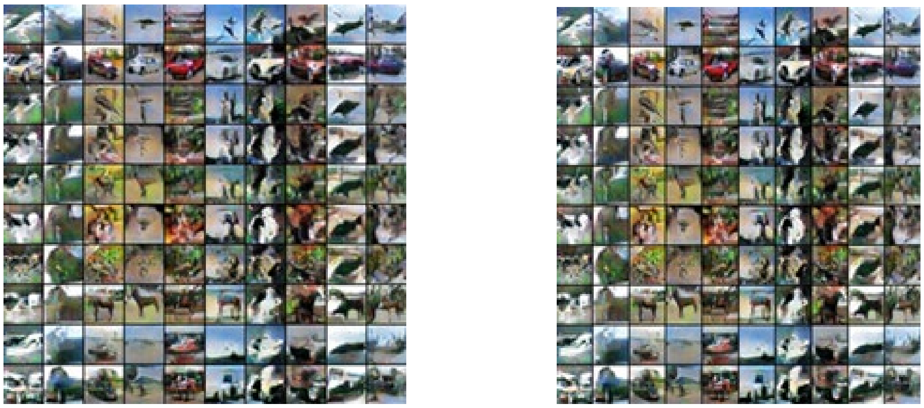
Figure 2: MINIST iterative training image effect of improved Triple-GAN model



(a) The 10th iteration (b) The 100th iteration (c) The 200th iteration

Figure 3: MINIST iteration training image effect based on Triple-GAN model

(2) From the analysis of image classification effects, from the comparative analysis of the classification effect of the cifar10 data set in Fig. 4 and the comparison of the classification effect of the cifar100 dataset in Fig. 5, it can be seen that the cifar10 and cifar100 data sets are based on the Improved Triple-GAN model and are better than Triple- The classification effect of GAN model.



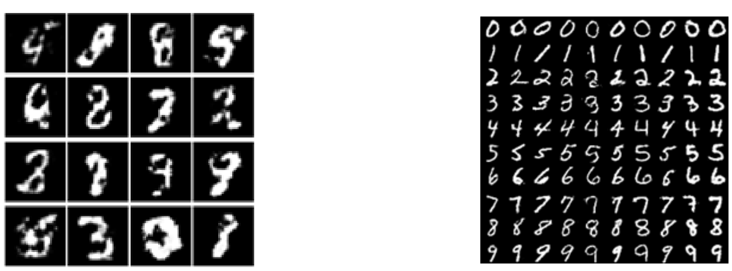
(a) Triple-GAN (b) Improved Triple-GAN

Figure 4: Comparison of cifar10 dataset classification results

(a) Triple-GAN (b) Improved Triple-GAN

Figure 5: Comparison of cifar100 dataset classification results

(3) From the image classification effect analysis, from the comparative analysis of the MINIST dataset in Fig. 6, it can be seen that the effect of the MINIST data set on the Improved Triple-GAN model is better than that of the LSGAN model, and the image is clearer.



(a) LSGAN (b) Improved Triple-GAN

Figure 6: Comparison of MINIST Data Set Sharpness

3.3 Results and Discussion

From the perspective of random forest algorithm and loss function obeying $\chi^2_{Pearson}$ distribution, the KL divergence is based on log (log) loss function, and $\chi^2_{Pearson}$ distribution is based on square (sqrt) form of distributed loss function. The distribution and random forest interface are used for effect analysis, as shown in Fig. 7. The results show that with the increase of the number of iterations, the error rate of random forest and $\chi^2_{Pearson}$ distribution is significantly lower than that of KL-divergence.

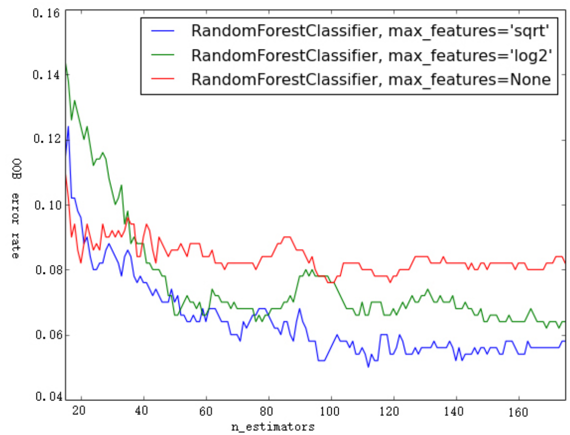


Figure 7: Random forest classifier adaptive training analysis graph

In the Triple-GAN optimization model, the generator, discriminator, and classifier loss crossover curves of the loss function are analyzed. As shown in Fig. 8, the crossover loss is distributed between 2.5% and 5%. The rate remained relatively low, indicating that the Improved Triple-GAN model is relatively stable.

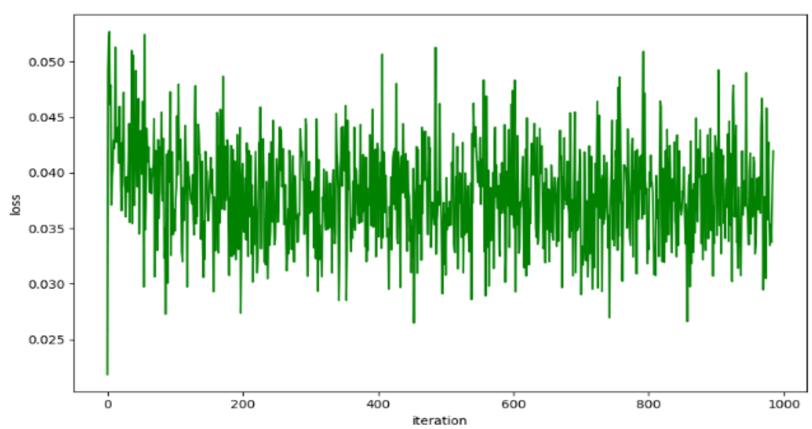


Figure 8: Loss function cross-loss distribution of Improved Triple-GAN model

From clarity and classification effect, the training data of the datasets of MINIST are more ideal. From the output data of the training process, the pre-training accuracy rate of different training numbers is extracted, as shown in Fig. 9. From the pre-training training accuracy curve in Fig. 4, it can be seen that as the number of pre-training increases, the accuracy of randomly arranged data and tag classification continues to decrease; as the number of training increases, the final accuracy tends to be one. The balance is stable at about 2.75%. From the perspective of classification performance, MINIST's simple hand-written digit set is taken as an example. Experiments show that MINIST running on the Improved Triple-GAN model can be quickly classified, and it can have a clear classification effect through iterative training.

The Improved Triple-GAN model generates a sample effect for the generator, as shown in the randomly generated visual matching error rate graph of Fig. 10. It can be seen that with time, the generation error rate is significantly reduced, indicating that the random forest algorithm generates a similar direction for the generator and generates similar data sets in the data set class, and the classification is made smaller and smaller, making the generation the sample generation error rate is getting smaller and smaller. From the beginning 2.8% to 1.2%, which greatly reduces the error rate.

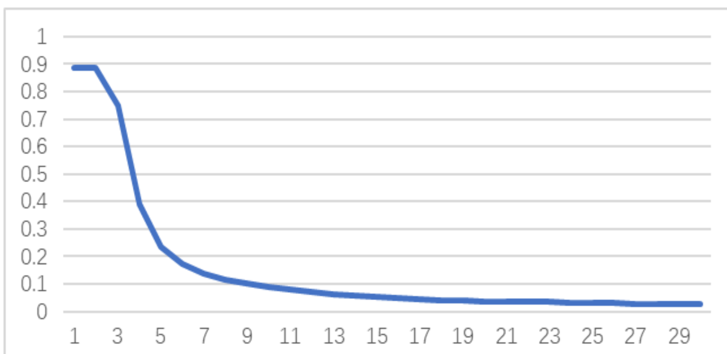


Figure 9: Pre-training accuracy curve

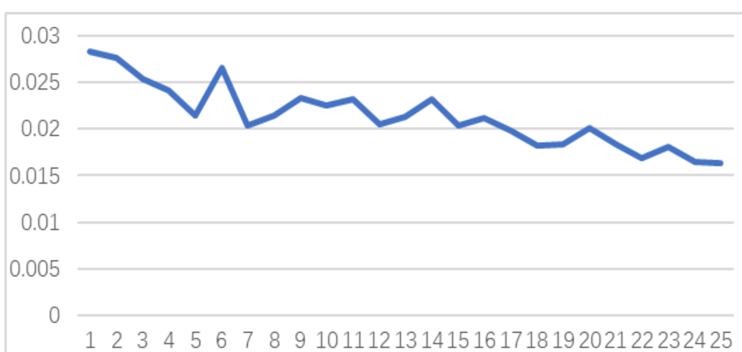


Figure 10: Randomly generated visual match error rate graph

4 Conclusions

Based on the above experiments, it can be seen that the Improved Triple-GAN model classifies and sorts the same sort of insertions, which greatly reduces the error rate of generated samples. At the same time, the data set classification effect and sample definition are greatly improved. And there is a great improvement in model training stability and label tag automation.

Since there are few kinds of datasets, more datasets will be used to verify the practicality of the Improved Triple-GAN model and the reliability of the model will be evaluated in terms of speed, effect, and performance. At the same time, the Improved Triple-GAN model can also be combined with the principle of CNN to continuously supervise the learning classification. The generated sample data can be continuously learned by CNN, similar data can be generated, and duplicate generation or overfitting generation can be avoided, thereby greatly improving the speed of the algorithm.

Funding Statement: This work was supported by Hengyang Normal University Hunan Province Key Laboratory for Digital Technology and Application of Settlement Cultural Heritage Open Fund Project "Based on CNN-based 3D Image Reconstruction Research" (JL16K05), Ministry of Science and Technology of the People's Republic of China (2018AAA0102301).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Adele, C.; Richard Cutler, D.; John Stevens, R. (2011): Random forests. *Ensemble Machine Learning*, pp. 157-175

Alec, R.; Luke, M.; Soumith, C. (2015): Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv.org, arXiv*:1511.06434.

Amorim, R. C. D.; Mirkin, B. (2012): Minkowski metric, feature weighting and anomalous cluster initializing in K-Means clustering. *Pattern Recognition*, vol. 45, no. 3, pp. 1061-1075.

Arjovsky, M.; Bottou, L. (2017): Towards principled methods for training generative adversarial networks. *International Conference on Learning Representations*.

Brock, A.; Donahue, J.; Simonyan, K. (2019): Large scale GAN training for high fidelity natural image synthesis. *arXiv.org, arXiv*:1809.11096.

CIFAR-10 Dataset (2019): http://www.cs.toronto.edu/~kriz/cifar-10-python.tar.gz, Last accessed on September 10.

CIFAR-100 Dataset (2019): http://www.cs.toronto.edu/~kriz/cifar-100-python.tar.gz, Last accessed in September.

Du Plessis, M. C.; Niu, G.; Sugiyama, M. (2015): Convex formulation for learning from positive and unlabeled data. *ICML*, vol. 37, pp. 1386-1394.

Jitender, E.; Julka, E. N. (2017): An approach for disease data classification using fuzzy support vector machine. *IOSR Journal of Electronics and Communication Engineering*, vol. 12, no. 1, pp. 1-7.

Leonardo, C.; Alastair, B. (1998): A comparison of batch and incremental supervised learning algorithms. *European Symposium on Principles of Data Mining and Knowledge Discovery*, pp. 264-272.

Li, C.; Xu, K.; Zhu, J.; Zhang B. (2017): Triple generative adversarial nets. *Conference on Neural Information Processing System*.

Liu, D.; Fan, S. (2014): A modified decision tree algorithm based on genetic algorithm for mobile user classification problem. *The Scientific World Journal*: 468324-468324.

Ma, J.; Yu, W.; Liang P.; Li, C.; Jiang, J. (2019): FusionGAN: a generative adversarial network for infrared and visible image fusion. *Information Fusion*, vol. 48, pp. 11-26.

Mao, X.; Li, Q.; Xie, H. (2017): Least squares generative adversarial networks. *2017 IEEE International Conference on Computer Vision*, vol. 1, pp. 2813-2821

MNIST Dataset (2019): http://www.iro.umontreal.ca/~lisa/deep/data/mnist/mnist.pkl.gz, Last accessed on January.

Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A. et al. (2016): Improved techniques for training GANs. *Advances in Neural Information Processing Systems*, pp. 2226-2234.

Sutherland, D. J.; Tung, H.; Strathmann, H.; De, S.; Ramdas, A. et al. (2017): Generative models and model criticism via optimized maximum mean discrepancy. *International Conference on Learning Representations*.

Wang, C.; Xu, C.; Yao, X.; Tao, D. (2019): Evolutionary generative adversarial networks. *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 6, pp. 921-934.

Zhang, H.; Goodfellow, I.; Metaxas, D.; Odena, A. (2018): Self-attention generative adversarial networks. *arXiv.org, arXiv*:1805.08318.

Zhou, Z.; Liang, J.; Song, Y.; Yu, L. T.; Wang, H. W. et al. (2019): Lipschitz generative adversarial nets. *arXiv.org, arXiv*:1902.05687.