Transactions on Computer Science and Intelligent Systems Research
ISSN: 2960-1800, eISSN: 2960-2238 | Volume 5
AIDML 2024

Exploiting Deep Convolutional Generative Adversarial Network Generated Images for Enhanced Image Classification

Jin Xing *

College of General Studies, Southern University of Science and Technology, Shenzhen, Guangdong, 518000, China

* Corresponding Author Email: 12210433@mail.sustech.edu.cn

Abstract. The power of deep neural networks relies heavily on the quantity and quality of training data. However, it is expensive and time consuming for people to collect and annotate data on a large scale. Traditional methods, including modifying the copies of existing data, do not always have the effect, especially in some biomedical fields where some large-size anonymous datasets are generally not publicly available. So, this paper tried to tackle this problem by generating specific data using Deep Convolutional Generative Adversarial Network (DCGAN). DCGAN structure combines convolution and traditional generative adversarial network, has the advantages of producing the clearer images than vanilla Generative adversarial network (GAN). The training dataset is from CIFAR-10 dataset, consist of 10 classes of natural item images. To measure whether it is useful, three classifiers, LeNet, AlexNet and InceptionNet, are trained by feeding original dataset and original dataset mixed with generated data. The final result is presented by comparing accuracy. It goes well by adding more generated data from DCGAN into the original data. The result proves that DCGAN is able to augment data.

Keywords: Deep learning; generative adversarial network; image augmentation.

1. Introduction

When conducting experiments such as analysis and summary, effective and large amounts of data are often required. By summarizing a large amount of data, the result can be more objective and accurate and avoid the accidental error. However, in the process of data collection, due to the service life of the collection device, collection errors or human interference, it is difficult to gain sufficient amounts of effective data in many cases [1,2]. In addition, in fields such as biomedicine. In order to protect patient privacy, many data cannot be open and to demonstrate to the public. In the above-mentioned cases, it is a waste of time and resources to continue to supplement and collect data. At this time, it is extremely important to use generative models to get accurate data to achieve the dataset augment.

Generative adversarial network (GAN), an effective generative deep learning model, is also a perfect method to extend the dataset. GAN uses the generator network to generates data and judge whether they are real or not through the discriminator network [3]. After a series of generating and judging procedures, GAN has built a well-developed model for generating data. There are many further developments on GAN later. As a variant update of vanilla GAN, Deep Convolutional GAN (DCGAN) uses structures like transposed convolution or batch normalization (BN) to make it more stable and avoid crashes during training [4].

As for the augment of image dataset, traditional methods are flip, rotation, scale, crop or translation in deep learning. In 2022, DALL-E2 and Stable Diffusion (SD) models are used to construct the Guided Imagination Framework, and then generate realistic images with new content [5]. There are also some relevant researchers combining transformers with GAN architectures to build a new strong GAN that do not contain convolutions at all [6]. Researchers also created a teacher-student GAN (TSGAN) to improve face-recognition, which consists of a teacher, with one generator and one discriminator, and a student, with two generators in the form of encoder-decoders and one discriminator [7].

In this paper, DCGAN is used to extend the dataset., taking CIFAR-10 for example. The specific code implementation uses the pytorch framework. The generated results are test by LeNet [8], AlexNet [9] and InceptionNet V1 [10], three verifiable classifiers. These classifiers are also trained by the original data from CIFAR-10 so that the augmented dataset can be discriminated and obtain their accuracy.

2. Method

2.1. Dataset

The dataset used in this paper is the CIFAR-10 dataset, from www.cs.toronto.edu/~kriz/cifar.html. It is labeled subsets of the 80 million tiny images dataset. There are 10 categories of RGB color images: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck. The pixel size of every image is 32 × 32, and there are 6,000 images per category, with a total of 50,000 training images and 10,000 test images in the CIFAR-10 dataset. In contrast to handwritten characters, the CIFAR-10 dataset includes noisy real-world objects whose proportions and properties vary greatly, making recognition extremely challenging. Therefore, in CIFAR-10 dataset, direct linear models like Softmax did not perform well.

2.2. Model

2.2.1. Generative Neural Network (GAN)

The inspiration behind GAN models is derived from the concept of zero-sum games, consisting of two main components: the generative model G and the discriminative model D. The generative model captures the distribution of sample data, while the discriminative model distinguishes whether the input is real data or generated samples. The following is the optimization function formula for GAN.

$$\min_G \max_D V(D,G) = \mathbb{E}_{X\sim P_{data}(x)}[logD(x)] + \mathbb{E}_{z\sim P_z(z)}[\log(1 - D(G(z)))] \quad (1)$$

X is the real data, and the real data conforms to the $P_{data}(x)$ distribution. z is noise data, and noise data conforms to the $P_z(z)$ distribution. Sample from z and generate x=G(z) through G. D will increase V(D, G) while G will decrease V(D,G).

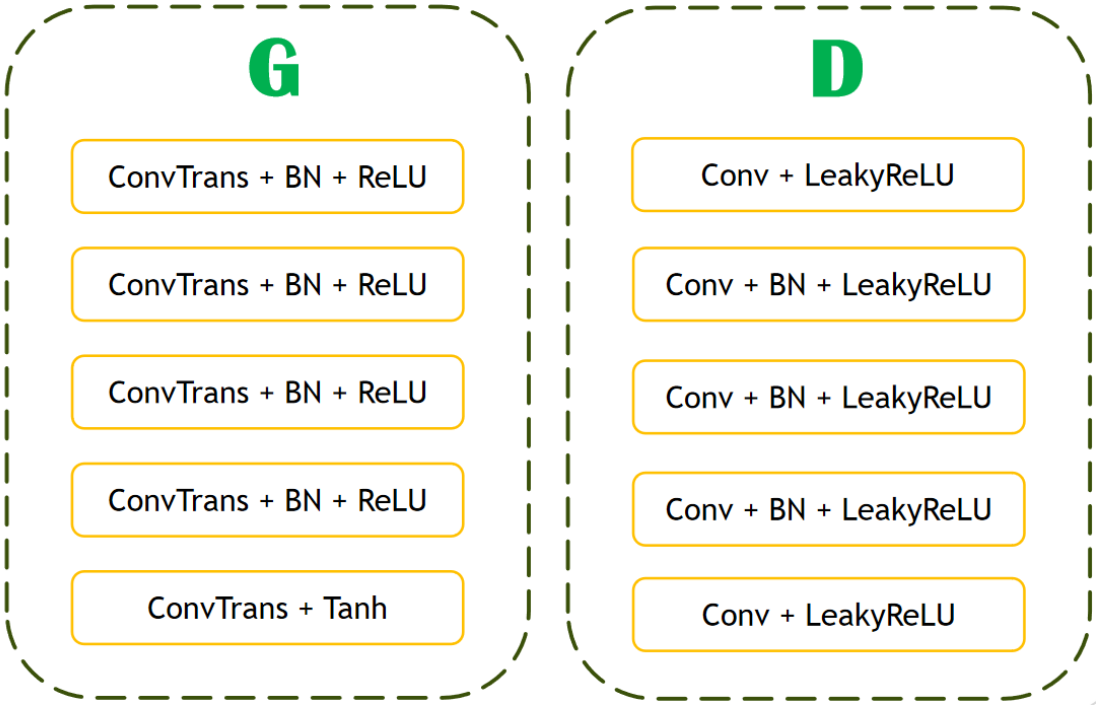

Fig. 1 Architecture of generator and discriminator (Figure Credits: Original).

2.2.2. Deep Convolutional Generative Neural Network (DCGAN)

Compared with vanilla GAN, the following changes were made to the DCGAN model. The specific architecture of the generator and discriminator are demonstrated in Fig. 1. First, the full connected layers are replaced by transposed convolution since full connection has a long-term training due to too much parameters and it exists a high possibility of over-fitting. The kernel size of transposed convolution layers is 4x4, with stride is 2x2 and 1x1 padding except for the first transposed convolution's 1x1 stride and no padding. Then, batch normalization is used after the first three transposed convolution, which is used to stabilize the training and avoid model crash easily. If BN is applied in all layers, training may still result in instability, so it is canceled at the last layer. Followed by the BN is the activation function. The activation functions in the two network is not same as well. In the generator, the activation function is ReLU while the last one is Tanh. In the discriminator, every activation function uses LeakyReLU. As for sigmoid function, when the input is too large or too small, the function gradient is almost 0, which is not conducive to backpropagation, the mean of Sigmoid is not 0, so that only all positive or all negative feedback can be generated during the network training. Tanh solves the problem that the mean value of Sigmoid function is not 0, and Tanh function is usually better than Sigmoid function. Besides, ReLU can converge the network more easily than Sigmoid and Tanh because of its special structure. But ReLU causes some neurons will never renew since it does not have the negative part, so LeakyReLU is used to mitigate the disappearance of this gradient.

2.2.3. Classifier

LeNet was proposed in 1998, which was one of the earliest convolutional neural networks [8]. At first it was used on handwritten character recognition. It consists of three main modules. The first and second modules have 5x5 convolution layers and 2x2 pooling operation. The first convolutional layer only has 6 output channels, while the second has 16 channels. Through spatial downsampling, each 2x2 pooling operation (with stride 2) reduces dimensionality by a factor of 4. The third module has 5x5 convolution layer with 120 filters. Then the feature extracted after these third convolution. The number of output neurons of the first fully connected layer is 64, and the next is the number of categories of the classification label, which is 10 for handwritten digit recognition. The Softmax activation function is then used to calculate the predicted probability for each category, since ReLU had not yet been discovered.As an ancient neural network, its accuracy usually demonstrates a little low.

AlexNet was released in 2012, and it has greatly stimulated the industry's interest in neural networks [9]. Compared with LeNet, it has a deeper network structure. AlexNet has five convolution layers and three full-connected layers. In AlexNet's first layer, the convolution window shape is 11x11. The convolution window shape in the second layer is reduced to 5x5, followed by 3x3. In addition, after the first, second, and fifth convolution layers, the network adds max-pooling layers with a shape of 3x3 and a stride of 2. After the final convolution layer, there are two huge fully connected layers with totally 4096 outputs. Duo to the early GPUs had limited memory, AlexNet's original architecture used a dual data stream approach, allowing each of its two GPUs to be in charge of only storing and processing half of the model. A Dropout layer is added behind each full-connected layer to suppress overfitting. And finally, a 1000 filter full-connected layer is applied. At this time, ReLU function is used to avoid the vanishing gradient problem.

In the 2014 ImageNet competition, InceptionNet is the major structure of the first price GoogleNet, and the network's main feature is that it not only has depth, but also has "width" horizontally [10]. Inception module adopts the design form of multi-path. Every branch uses convolution kernel of different sizes. Besides, the number of channels in the final output feature map is the sum of the output channels from each branch. In order to reduce the number of parameters, the Inception module actually adds 1x1 convolution layers before each 3x3 and 5x5 convolution layer to adjust the number of output channels, and adds 1x1 convolution layers behind the maximum pooling layer to reduce the number of output channels.

3. Result

The DCGAN model has been train for 50 epochs, with 50000 images for training and 10000 for testing from CIFAR-10 dataset. The generation was from basic random noise. The images in Fig. 2 demonstrate the effect of DCGAN's generation.

Fig. 2 Representative examples of generated images (Figure Credits: Original).

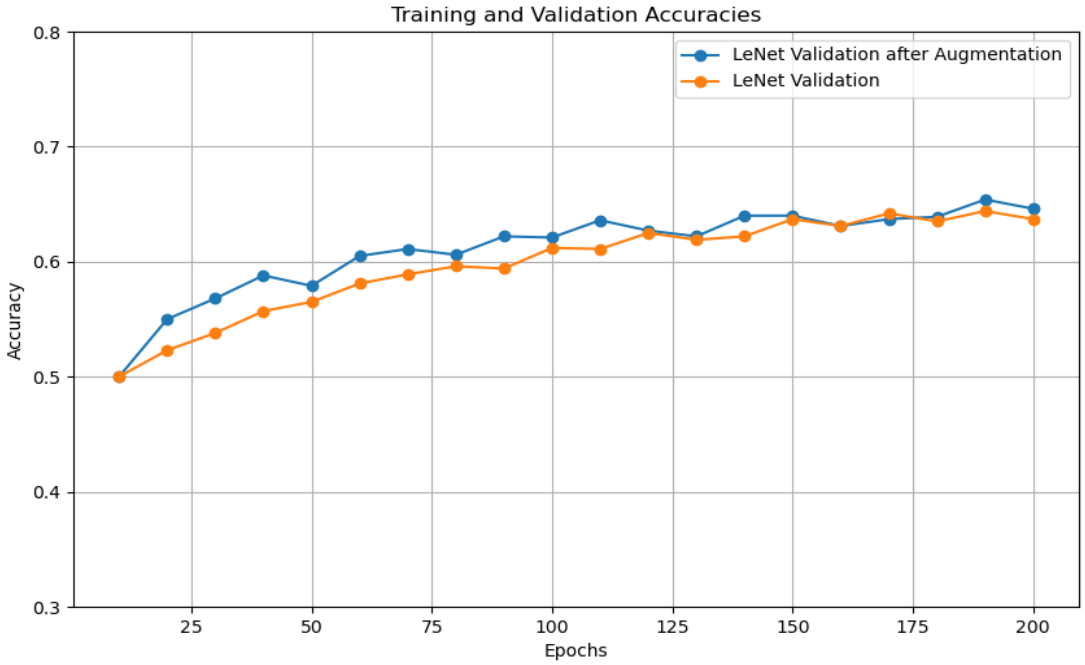

Fig. 3 Accuracy on LeNet with and without DCGAN augmented images (Figure Credits: Original).

After training the DCGAN model, this work generated 3200 images for each class, leading to a total of 32,000 images. Subsequently, this work merged these newly synthesized data with the original CIFAR-10 dataset consisting of 50,000 images to create an augmented dataset comprising a total of 84,000 images. This work then proceeded to train the LeNet, AlexNet, and InceptionNet classifiers on both the original dataset (50,000 images) and the augmented dataset (84,000 images) containing DCGAN-generated images. A shared test dataset of 10,000 images was used for evaluation purposes. Due to LeNet's comparatively slower convergence rate in comparison to the other two classifiers, it underwent training for 200 epochs while the remaining two were trained for 100 epochs each. Accuracy was calculated and recorded throughout the training process as a visual metric for comparative analysis using Python's matplotlib library.

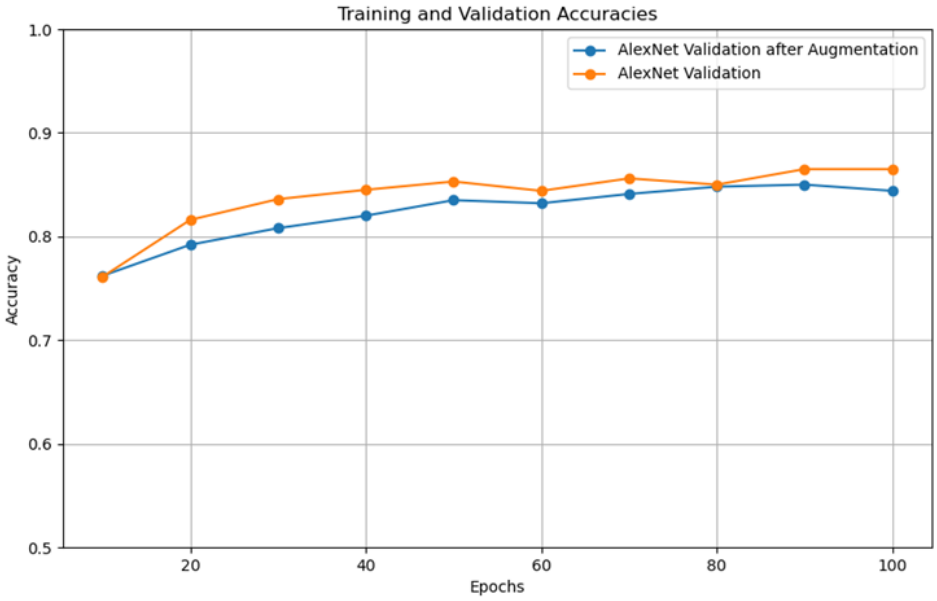

Fig. 4 Accuracy on AlexNet with and without DCGAN augmented images (Figure Credits: Original).

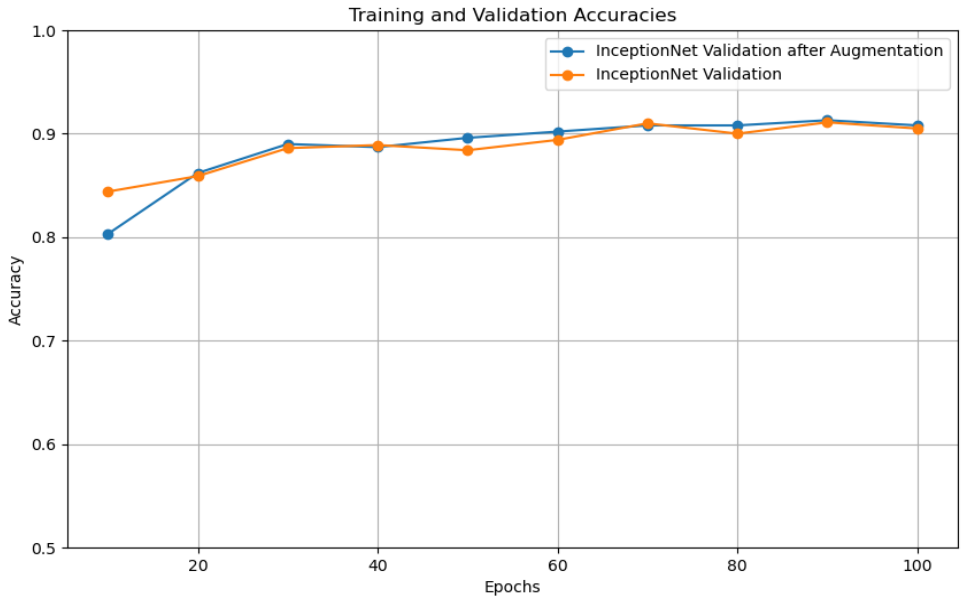

Fig. 5 Accuracy on InceptionNet with and without DCGAN augmented images (Figure Credits: Original).

From Fig. 3, Fig. 4, and Fig. 5, results shown that although the final values are not exactly the same, they all converge to a relatively stable range. This indicates that the results of this experiment are reliable. A comparison between the original dataset and the augmented dataset shows that the differences fall within a reasonable margin of error, demonstrating that the augmented dataset is accurate and does not introduce significant biases due to augmentation. This overall result highlights the good data augmentation capability of DCGAN.

4. Discussion

Obviously, DCGAN still has some shortcomings and is not currently the most optimal model. The generation time of DCGAN is relatively long, and there seem to be some unnatural details in some parts of the pseudo data which are similar to the original images. While it can still generate clearer and more stable images compared to vanilla GAN. However, the advantage of DCGAN is its model's

high flexibility, allowing for further optimization of the generated quality by adjusting the number of layers, structure, or loss function. Based on experience after conducting more experiments and generating data in the certain field, more improvement can be applied on this basic model of DCGAN, and some difficulty of data augmentation will be solved.

In terms of data, the pixel size of CIFAR-10 data is relatively low, and switching to high-definition images with higher precision requirements in the future may lead to a decrease in accuracy. According to the experimental results of this study, the accurate convergence range of the final accuracy cannot be precisely determined, which can be improved by increasing the training epochs. Incorporating metrics, like recall, precision, and F1 score, can comprehensively evaluate the quality of data augmentation in the dataset. Simultaneously, more advanced classifiers, such as ResNet, can also be used to further explore the accuracy of data augmentation.

5. Conclusion

To sum up, to achieve the effect of dataset augmentation from DCGAN, this work uses the CIFAR-10 dataset as origin dataset to train. After generating 32000 pseudo images and merging them into origin dataset, LeNet, AlexNet and InceptionNet are used to justify the accuracy of generation quality. Finally, from the line charts, the result is that the both accuracies can be converted to a high level, which proves the rationality of experiment. The major idea of this paper is to augment the data hard to access, it has bright application on many fields with developed structure of generative network.

References

[1] Hemkens, Lars G., et al. The reporting of studies using routinely collected health data was often insufficient. Journal of clinical epidemiology, 2016, 79: 104-111.

[2] Yang, Xiangli, Song, Zixing, et al. A survey on deep semi-supervised learning. IEEE Transactions on Knowledge and Data Engineering, 2022, 35(9): 8934-8954.

[3] Goodfellow, Ian and Pouget-Abadie, Jean and Mirza, et al. Generative adversarial nets. Advances in neural information processing systems, 2014, 27: 1-9.

[4] Radford, Alec, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. ArXiv Preprint, 2015: 1511.06434.

[5] Zhang, Yifan, Zhou, Daquan, et al. Expanding small-scale datasets with guided imagination. Advances in Neural Information Processing Systems, 2024, 36: 1-61.

[6] Jiang, Yifan, Shiyu Chang, and Zhangyang Wang. Transgan: Two pure transformers can make one strong gan, and that can scale up. Advances in Neural Information Processing Systems, 2021, 34: 14745-14758.

[7] Uppal, Hardik, Sepas-Moghaddam, Alireza, et al. Teacher-student adversarial depth hallucination to improve face recognition. Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 3671--3680.

[8] LeCun, Yann, et al. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[9] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 2012, 25: 1-9.

[10] Szegedy, Christian, et al. Going deeper with convolutions. Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 1-9.