Diversifying Image Synthesis using Data Classification

Yuta Suzuki, Fumihiko Sakaue and Jun Sato
Nagoya Institute of Technology, Japan

Keywords: Diverse Image, Data Classification, GAN.

Abstract: In this paper, we propose a method for generating highly diverse images in GAN-based image generation. In recent years, GANs that generate various images such as MSGAN and BicycleGAN have been proposed. By using these methods, it is possible to generate a variety of images to some extent, but when compared with the variety of training images, they are still less diverse. That is, it is still a difficult problem to generate a variety of images, even if a wide variety of training images are being trained. Thus, in this paper, we propose a new structure of GAN which enables us to generate more diversity than the existing methods. Our method estimates the distribution of training images in advance and learns to imitate the diversity of training images. The effectiveness of the proposed method is shown by comparative experiments with the existing methods.

1 INTRODUCTION

In recent years, research on GAN (Goodfellow et al., 2014), which generates highly realistic images by deep learning, has been progressing. While general GAN can generate a realistic image from random noise, conditional GAN (cGAN) (Mirza and Osindero, 2014; Isola et al., 2017) can generate images according to the given labels.

In many conditional GAN studies, it was important to output realistic images from labels, and the diversity of generated images was not so important. However, in many image generation tasks, there are many optimal images for a single label. For example, in the task of ”edges-to-shoes”, we human can imagine various kinds of shoes such as red shoes and blue shoes from a single edge image as shown in Fig. 1. However, when we use pix2pix (Isola et al., 2017) for this task, it is not possible to generate images with such variety of colors and shapes.

On the other hand, in recent years, some new types of GANs, such as BicycleGAN (Zhu et al., 2017) and MSGAN (Mao et al., 2019), succeeded in diversifying the generated images by incorporating losses which evaluate the diversity of generated images. However, even in these GANs, the diversity of generated images is still not high compared to the diversity of training images.

Therefore, in this paper, we propose a method for generating a variety of images equal to or greater than the variety of training images. In the existing methods, the network training was conducted without knowing the distribution of training images. On the other hand, in our method, we estimate the distribution of training images in advance, and learn to imitate the diversity of training images. For estimating the distribution of training images, we perform k-means clustering of the training images by using feature vectors extracted from a pre-trained classification network. In this way, our GAN can generate a wide variety of images that are closer to the ground truth distribution of training images.

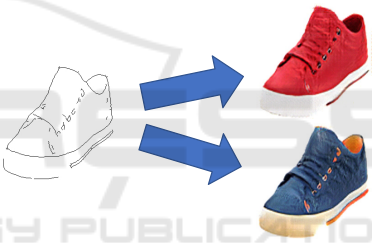

Figure 1: Generation of diverse images.

We perform a comparative experiment between the proposed method and the existing methods using quantitative evaluation, and show that the proposed method can generate a wider variety of images.

2 RELATED WORK

GAN (Goodfellow et al., 2014) is a learning model in which the Generator that generates images and the

Suzuki, Y., Sakaue, F. and Sato, J.
Diversifying Image Synthesis using Data Classification.
DOI: 10.5220/0010902600003124
In *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2022) - Volume 5: VISAPP*, pages 617-622
ISBN: 978-989-758-555-5; ISSN: 2184-4321

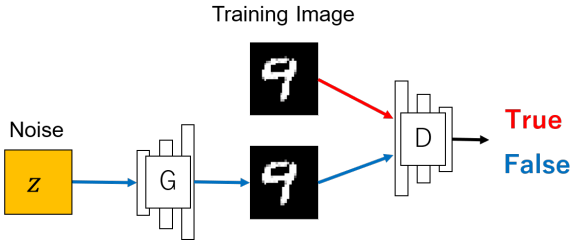

Figure 2: Structure of general GANs.

Discriminator that distinguishes between generated data and training data perform adversarial training to enhance each other's performance as shown in Fig. 2.

Generator tries to generate data similar to the training data, and learns to generate images that fool Discriminator. On the other hand, Discriminator tries to distinguish the authenticity of the input data, that is Discriminator determines that the input data is True if it is training data, and False if it is generated by the Generator. By training Generator and Discriminator adversarially, the Generator will gradually be able to generate realistic images, and finally the Generator will be able to generate data that cannot be distinguished by the Discriminator. GAN has been actively researched in recent years since it can generate images with high accuracy that is indistinguishable from the real images.

While general GAN can generate a realistic image from random noise, conditional GAN (cGAN) (Mirza and Osindero, 2014; Isola et al., 2017) can generate images according to the given labels. However, in these GANs, the reality of images was often emphasized, and the diversity of generated images was not considered much.

On the other hand, MSGAN (Mao et al., 2019) challenged the diversification of generated images. As shown in Fig. 3, the training of MSGAN is performed so that the difference d_I of output $G(y,z_1)$ and $G(y,z_2)$ generated from two random noise z_1 and z_2 becomes large even if the difference $d_z(z_1,z_2)$ between z_1 and z_2 is small by adding a new regularization term called MS loss. The MS loss $\mathcal{L}_{MS}$ is a ratio between the distance d_z of the input noises and the distance d_I of the generated images as follows:

$$\mathcal{L}_{MS} = \max_G \frac{d_I(G(y,z_1),G(y,z_2))}{d_z(z_1,z_2)} \quad (1)$$

The total loss $\mathcal{L}_{MSGAN}$ of MSGAN is defined by adding the MS Loss $\mathcal{L}_{MS}$ to the loss of original GAN $\mathcal{L}_{ori}$ as follows:

$$\mathcal{L}_{MSGAN} = \mathcal{L}_{ori} + \lambda_{MS}\mathcal{L}_{MS} \quad (2)$$

where, λ_{MS} represents the weight of the MS Loss.

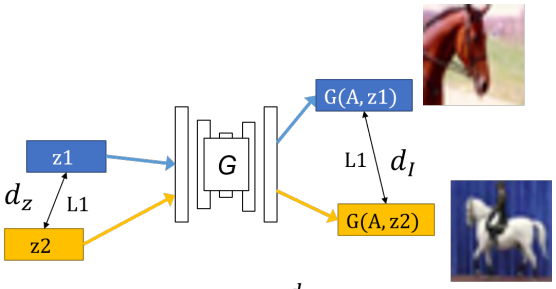

Figure 3: Training of MSGAN.

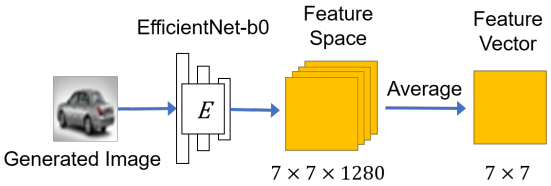

Figure 4: Feature extraction.

Although MSGAN succeeded in improving the diversity, it is still not sufficiently high compared to the diversity of training images.

3 PROPOSED METHOD

In this paper, we propose a method for generating images with diversity close to that of the training images. The proposed method estimates the distribution of the training images in advance, and learns so that the distribution of the generated images is close to the distribution of the training images.

3.1 Estimation of Training Image Distribution

In our method, the distribution of training images is estimated first by using k-means clustering before training GAN.

For this objective, a pre-trained classification network is used as an encoder to extract the feature vector of the image. In this research, the feature vector is extracted by using Efficient-Net-b0 (Tan and Le, 2019), which had been pre-trained with ImageNet (Deng et al., 2009). We first extract $7 \times 7 \times 1280$ features by using Efficient-Net-b0, and then the obtained features are averaged for each channel and a $7 \times 7 = 49$ dimensional feature vector is obtained as shown in Fig. 4.

Then, k-means clustering of the training images is performed by using the extracted feature vectors, and

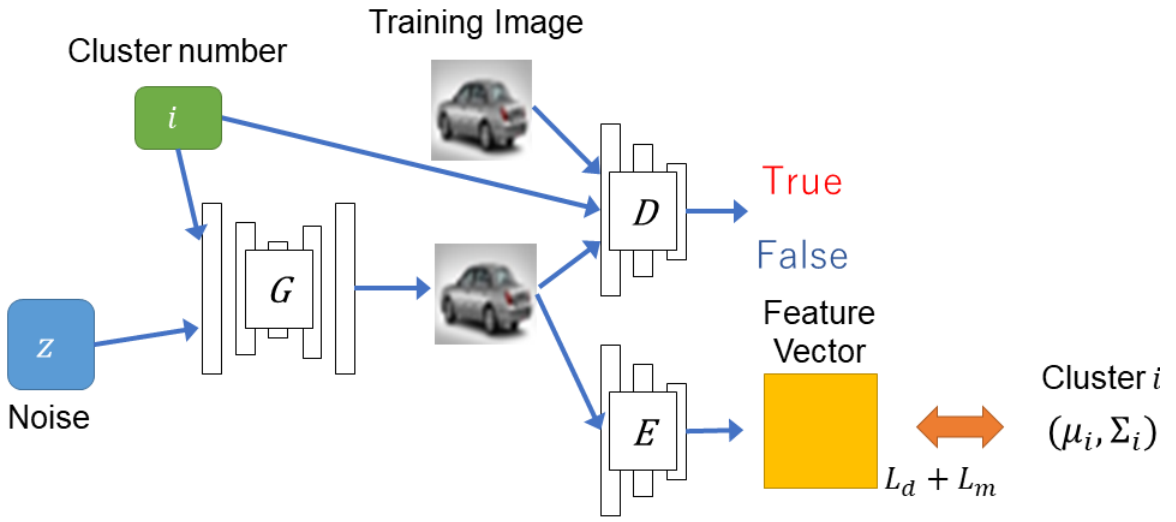

Figure 5: Network structure of the proposed method. G, D, and E represent Generator, Discriminator, and Encoder respectively. G and D are similar to the conventional GAN, and Encoder is a pre-learned feature extractor. In the proposed method, a part for extracting the features of the generated image is added. We compute the loss by comparing the mean μ and the covariance matrix Σ of the extracted features with those $\{\mu_i, \Sigma_i\}$ of cluster i.

the training images are classified into k clusters.

The distribution of the training images is estimated by computing the mean vector μ_i $(i = 1, \cdots, k)$ and the covariance matrix Σ_i $(i = 1, \cdots, k)$ of feature vectors in each cluster.

In the following sections, we describe the GAN network structure and learning method using the estimated distribution of training images.

3.2 Network Structure

In this research, we input a cluster number to Generator, and the Generator generate images with different characteristics according to the input cluster number. This enables us to generate a variety of images from a single Generator.

For this objective, we add the cluster number i to the input of the conventional GAN as shown in Fig. 5. The cluster number is also input to the Discriminator, and the Discriminator determines not only whether the image is real or not, but also whether the image that matches the cluster number can be generated or not.

E in Fig. 5 is a pre-trained feature extractor. The training of the network is performed so that the feature vector obtained through this feature extractor approaches the feature vector of the image of cluster i.

The proposed network structure can be applied to various tasks, such as image2image and text2image. In these cases, not only the cluster number and noise but also the condition image and condition text are added to the input of Generator and Discriminator.

3.3 Training

At the training stage, the cluster number to which the training image belongs is input to the Generator, and the Generator is trained so as to generate images with the characteristics of the cluster.

Discriminator judges whether the Generator is able to generate an image that matches the input cluster number. However, the generated clusters are not as easily distinguishable by humans as dog clusters and car clusters. Therefore, in order to determine whether the generated image includes the characteristics of the cluster, the following two losses are introduced for training the network.

$$\mathcal{L}_d = |\mu_i - E(G(z,i))| \tag{3}$$

$$\mathcal{L}_m = \sqrt{(E(G(z,i)) - \mu_i)^\top \Sigma_i^{-1} (E(G(z,i)) - \mu_i)} \tag{4}$$

where, μ_i is the mean feature vector of cluster i, and Σ_i is the covariance matrix of feature vectors of cluster i.

$\mathcal{L}_d$ is the L1 distance between the generated image and the center of cluster i, and $\mathcal{L}_m$ is the Mahalanobis distance between the generated image and cluster i. By using the Mahalanobis distance $\mathcal{L}_m$, the variance is taken into consideration and learning is performed so as to generate an image containing more cluster-specific features. The L1 distance $\mathcal{L}_d$ is used for stabilizing the training.

In this research, the above two losses are used with MS loss and original loss, so the total Loss is as follows:

$$\mathcal{L}_{new} = \mathcal{L}_{ori} + \lambda_{MS}\mathcal{L}_{MS} + \lambda_d\mathcal{L}_d + \lambda_m\mathcal{L}_m \tag{5}$$

Figure 6: Results of clustering the training images in the proposed method. The figure shows 7 example images classified into each of the 10 clusters. In the proposed method, the training was performed using the results of the clustering.

By using the MS Loss with $\mathcal{L}_m$ and $\mathcal{L}_d$, it is expected that diversity will be produced in the same cluster where similar images are gathered, and more diverse images can be generated.

Depending on the number of images and the number of clusters used for training, the number of features in the feature vector may be larger than the number of images n belonging to the smallest cluster. In such a case, the covariance matrix cannot be computed properly. Therefore, in this research, the variance of the feature vector in the training image is computed, and the top $n-1$ features with large variance are used to compute the Mahalanobis distance. This is because the feature with a larger variance in the training image can be considered as a more important feature that changes greatly depending on the image.

3.4 Image Generation

The trained generator can be used for generating images. In the proposed method, we need to enter the cluster number for generating images. At the training time, the cluster number of the training image is input, but at the testing time, the cluster number is randomly determined according to the cluster size ratio in the training images, so that images with a distribution similar to that of the training dataset can be generated. For example, if the ratio of cluster 1 is 12 % in the training dataset, cluster number 1 is selected and input to the Generator with a probability of 12 %. This makes it possible to generate images with a distribution close to the training dataset.

4 EXPERIMENTS

We next show the experimental results obtained from the proposed method. In our experiments, we generated images by using the pix2pix fasades dataset (Yu and Grauman, 2014; Zhu et al., 2016). First, learning was performed using training data, and then image generation and quantitative evaluation using test data were performed.

The proposed method was trained with the number of clusters $k = 10$, the weight $\lambda_{MS} = 1.0$, $\lambda_d = 5.0$, and $\lambda_m = 0.2$. With this setting, 400 epoch training was performed using 400 training data. Since the training becomes unstable when the Mahalanobis distance is used from the beginning, it is introduced from 100 epoch, where the training has progressed to some extent.

We next show the results of comparative evaluation between the proposed method and the existing

Figure 7: Generated images in the proposed method and existing methods. The proposed method can generate more diverse images than the existing methods. For the proposed method, the cluster number is also shown.

Figure 8: Generated images in the proposed method and existing methods. The proposed method can generate more diverse images than the existing methods. For the proposed method, the cluster number is also shown.

methods. Fig. 6 shows the result of clustering the training images in the proposed method. This figure shows 7 example images classified for each cluster. In the proposed method, the training was performed using the results of this clustering.

Fig. 7 and Fig. 8 show the results of image generation by the proposed method and the existing methods. For the proposed method, the entered cluster number is also shown in the figure. From these results, we find that the proposed method can generate more diverse images than the existing methods for both input images. In particular, MSGAN did not generate many dark-colored buildings, but the proposed method was able to generate dark orange and yellow building images, indicating that images with a wider variety could be generated.

4.1 Quantitative Evaluation

Finally, the results of quantitative evaluation using FID (Heusel et al., 2017), NDB, JSD (Richardson and Weiss, 2018), and LPIPS (Zhang et al., 2018) are shown. We first explain each evaluation metrics.

FID

Since FID is an index for measuring whether or not the distribution of features obtained by passing each of the generated image and the training image through Inception Net (Szegedy et al., 2015) is close, we use FID to evaluate the reality of the generated images.

NDB and JSD

NDB and JSD are indexes to measure whether the distribution of training images and the distribution of generated images are similar by using bin-based metrics. The training images are first clustered into bins by k-means clustering, and then the generated images are assigned to bins of nearest clusters. Then, the similarity of the cluster distribution of the training images and that of the generated images is measured. It can

Table 1: Quantitative evaluation.

	pix2pix	MSGAN	Ours
FID	98.23	**88.84**	92.27
NDB	11	11	**9**
JSD	0.0812	0.0559	**0.0300**
LPIPS	0.0621	0.3752	**0.4444**

be said that the lower these two indicators, the closer the diversity is to the real data.

LPIPS

LPIPS measures the average distance between images in the feature space. In this research, we evaluated the diversity by measuring the average of LPIPS between the generated images. It can be said that the higher the LPIPS value, the more successful the generation of diverse images.

Results

The table 1 shows the results of quantitative evaluation of the existing methods and the proposed method. From this table, we fined that the proposed method shows the best score in metrics except FID, and we find that the proposed method can generate diverse images close to the ground truth distribution. Regarding FID, although it has decreased in the proposed method, the degradation is small, and we find that the diversification was successful while maintaining the quality of the generated images in the proposed method.

5 CONCLUSION

In this research, we proposed a method for generating more diverse images in GAN. In particular, we proposed a method that estimates the distribution of training images in advance and uses it for learning to generate diverse images. We demonstrated its effectiveness by conducting comparative experiments with the existing methods. The results show that the proposed method can generate more diverse images efficiently.

REFERENCES

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Li, F.-F. (2009). Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. IEEE Computer Society.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. (2017). Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NIPS*.

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. *CVPR*.

Mao, Q., Lee, H.-Y., Tseng, H.-Y., Ma, S., and Yang, M.-H. (2019). Mode seeking generative adversarial networks for diverse image synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Mirza, M. and Osindero, S. (2014). Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*.

Richardson, E. and Weiss, Y. (2018). On gans and gmms. In *Advances in Neural Information Processing Systems*.

Szegedy, C., Wei Liu, Yangqing Jia, Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9.

Tan, M. and Le, Q. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In Chaudhuri, K. and Salakhutdinov, R., editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6105–6114. PMLR.

Yu, A. and Grauman, K. (2014). Fine-grained visual comparisons with local learning. In *Computer Vision and Pattern Recognition (CVPR)*.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*.

Zhu, J.-Y., Krähenbühl, P., Shechtman, E., and Efros, A. A. (2016). Generative visual manipulation on the natural image manifold. In *Proceedings of European Conference on Computer Vision (ECCV)*.

Zhu, J.-Y., Zhang, R., Pathak, D., Darrell, T., Efros, A. A., Wang, O., and Shechtman, E. (2017). Toward multimodal image-to-image translation. *NIPS*.