A deep data augmentation framework based on generative adversarial networks

Wang, Qiping; Luo, Ling; Xie, Haoran; Rao, Yanghui; Lau, Raymond Y.K.; Zhang, Detian

Published in:
Multimedia Tools and Applications

Published: 01/12/2022

Document Version:
Post-print, also known as Accepted Author Manuscript, Peer-reviewed or Author Final version

Published version (DOI):
10.1007/s11042-022-13476-w

Publication details:
Wang, Q., Luo, L., Xie, H., Rao, Y., Lau, R. Y. K., & Zhang, D. (2022). A deep data augmentation framework based on generative adversarial networks. *Multimedia Tools and Applications*, *81*(29), 42871–42887. Advance online publication. https://doi.org/10.1007/s11042-022-13476-w

Noname manuscript No.
(will be inserted by the editor)

A Deep Data Augmentation Framework based on Generative Adversarial Networks

Qiping Wang · Ling Luo · Haoran Xie* · Yanghui Rao · Raymond Y.K. Lau · Detian Zhang

Received: date / Accepted: date

Abstract In the process of training convolutional neural networks, the training data is often insufficient to obtain ideal performance and encounters the overfitting problem. To address this issue, traditional data augmentation (DA) techniques, which are designed manually based on empirical results, are often adopted in supervised learning. Essentially, traditional DA techniques are in the implicit form of feature engineering. The augmentation strategies should be designed carefully, for example, the distribution of augmented samples should be close to the original data distribution. Otherwise, it will reduce the performance on the test set. Instead of designing augmentation strategies manually, we propose to learn the data distribution directly. New samples can then be generated from the estimated data distribution. Specifically, a deep DA framework is proposed which consists of two neural networks. One is a generative adversarial network, which is used to learn the data distribution, and the other

Qiping Wang
East China Normal University, Shanghai, China
E-mail: qpwang@fem.ecnu.edu.cn

Ling Luo
Sun Yat-Sen University, Guangzhou, China
E-mail: luol26@mail2.sysu.edu.cn

*Corresponding Author: Haoran Xie
Lingnan University, Hong Kong SAR
E-mail: hrxie2@gmail.com

Yanghui Rao
Sun Yat-Sen University, Guangzhou, China
E-mail: raoyangh@mail.sysu.edu.cn

Raymond Y.K. Lau
City University of Hong Kong, Hong Kong SAR
E-mail: raylau@cityu.edu.hk

Detian Zhang
Soochow University, Suzhou, China
E-mail: detian.cs@gmail.com

one is a convolutional neural network classifier. We evaluate our model on a handwritten Chinese character dataset and a digit dataset, and the experimental results show it outperforms baseline methods including one manually well-designed DA method and two state-of-the-art DA methods.

Keywords Data augmentation · Convolutional neural networks · Generative adversarial networks

1 Introduction

The development of Convolutional Neural Networks (CNNs) has led to significant improvement in many computer vision tasks including image classification [21,19], detection [13,5], and segmentation [33,49]. Although CNNs have achieved great success in computer vision, there are some limitations of CNNs. One of the major limitations is that CNNs are unable to learn invariant features [14]. For example, CNNs are sensitive to translation, rotation, and scaling. One effective method to solve this problem is the technique of data augmentation, which helps CNNs to learn invariant features from augmented samples with abundant variations of the original samples. Almost all successful models such as AlexNet [27], VGG [45], and ResNet [19] adopt data augmentation before training.

Traditional data augmentation methods are manually specified, where some transformations applied to samples are designed empirically. The transformations should be designed carefully. Otherwise, it may lead to a negative impact on the performance if the distribution of transformed samples is different from the original data distribution. This problem can be addressed by learning the augmentation method from the training data instead of designing it manually. Recently, a pairwise learning scheme for data augmentation has been proposed in literature [18]. They tried to learn the transformation distribution from sample pairs. However, the pairwise learning scheme requires modeling the transformations, which are also specified manually. Moreover, the modeling capabilities of pairwise transformations are limited as it only contains the designed representation of transformations and can not cover all the possible transformations between samples.

In this work, we propose to automatically learn the data augmentation algorithms from the data. Our method is to first learn the data distribution via deep generative networks and then generate new samples from the estimated data distribution. Generative Adversarial Network (GAN) [15] is one of the most successful deep generative models, which can generate high-quality images on some datasets. GANs consists of a generator and a discriminator. The generator tries to generate samples as real as possible, while the discriminator aims to distinguish whether the samples are real or fake. However, applying GANs directly on some datasets (e.g., the Chinese character dataset) can not generate readable Chinese characters as Fig. 1 shows. This is because of the lack of a deterministic relationship between input and output [20]. We use

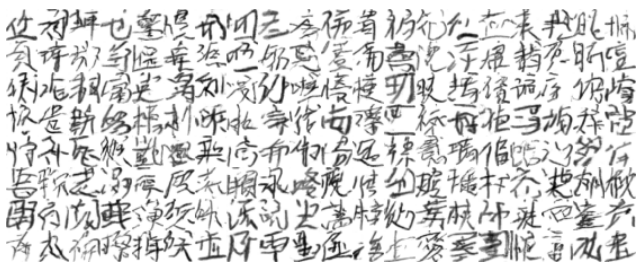

Fig. 1 Generated Chinese character images by regular GANs. These characters are unreadable.

conditional GANs to bridge the gap of the deterministic relationship between input and output, and this method can generate readable images.

Our main contribution in this paper is listed as follows.

- We propose a generic deep data augmentation (DDA) framework for convolutional neural networks based on GANs, and develop a corresponding training algorithm for the proposed DDA framework.
- We design the model architecture for implementing DDA framework for the conditional image generation.
- We evaluate the proposed framework on a Chinese character dataset and a digit dataset by comparing the baseline methods including one manually well-designed DA method and two state-of-the-art DA methods.

The remaining sections are organized as follows. Section 2 briefly reviews related studies about data augmentation and GANs. The proposed method is introduced in Section 3. Experiments on four datasets are presented in Section 4. In Section 5, we summarize the findings of this study and discuss possible future works.

2 Related Work

2.1 Data Augmentation

Data augmentation has become a necessary component for training convolutional neural networks. For example, Alexnet [27], VGG [45] and ResNet [19] all consist of image translation, horizontal reflection and altering intensities for data augmentation. In addition to these basic augmentation methods, many sophisticated methods have been proposed. Paulin *et al.* [40] proposed a method to automatically select the most useful transformations. In particular, the transformation that yields the highest accuracy gain is selected at each iteration. Another transformation selection strategy was proposed by [11], where the transformation that causes the maximal loss is selected. The intuition behind this strategy is to increase the diversity of the training images. Data augmentation is the key to high-dimensional model training. A lot

of data augmentation methods [53,50,30,8,48,35,52,23,31,10,7,51] have been proposed to improve the performance of deep CNN classification.

For the tasks of digit and character recognition, there are some domain-specific data augmentation methods [44,22,18]. Simard *et al.* [44] suggested applying elastic distortions for digit recognition. Jaderberg *et al.* [22] introduced various transformation methods such as projective distortion, blending, and JPEG compression and shown that it can train accurate models when only the synthetic data are used. The above methods are empirically designed and rely on manual specifications. Hauberg *et al.* [18] presented a data-driven data augmentation technique. They proposed to model the transformations as a distribution and inference it from the data. To generate new images, they sample a transformation from the learned distribution and then apply it to a randomly chosen image. Our proposed method is also data-driven. But instead of modeling the transformation, we inference the data distribution directly and then sample from the learned distribution to generate new images.

2.2 Generative Adversarial Networks

Generative Adversarial Networks (GANs) are one kind of the most successful generative models. Goodfellow *et al.* [15] first introduced the GANs in 2014. Then many GAN-based models have been proposed. Denton *et al.* [9] proposed Laplacian GAN which generates the images in a coarse-to-fine manner. Deep convolutional generative adversarial network (DCGAN) [42] is one of the important models, which incorporate the convolutional layers into GANs. DCGANs are capable of generating decent images on some scene datasets. Nowozin *et al.* [38] showed that any f-divergence can be used as the objective function to train GANs. Mao *et al.* [34] proposed to replace the traditional cross-entropy loss with the least squares loss, which can improve the generated image quality and the training stability. Arjovsky *et al.* [3] presented Wasserstein GAN that utilizes the Wasserstein distance to measure the distance between the real and fake distributions, and they proved that Wasserstein GAN can solve gradient vanishing problem in theory. Gulrajani *et al.* [16] incorporated the gradient penalty into Wasserstein GAN to further stabilize the training of Wasserstein GAN. Karras *et al.* [25] used a progressive training schema to stabilize GANs training, and their model can generate 1024×1024 high-fidelity images. Miyato *et al.* [37] proposed a weight normalization technique called spectral normalization to stabilize GANs training. Zhang *et al.* [17] proposed the self-attention GAN which utilizes the self-attention layer to improve the details of the generated images using cues from related locations. Brock *et al.* [4] showed that scaling up the model size can significantly improve the generated image quality for the challenging ImageNet dataset. Karras *et al.* [26] proposed a style-based architecture for the generator, which improves the generated image quality and better disentangles the latent factors.

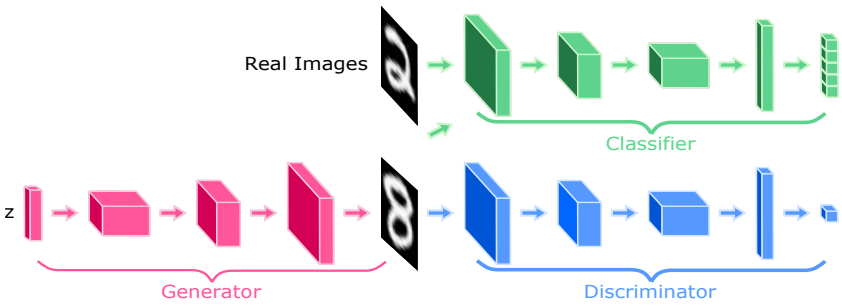

Fig. 2 The deep data augmentation framework

Several works focus on improving the performance of semi-supervised learning using GANs [43,39,46], where the discriminator is trained not only to distinguish real and fake images but also to classify the labeled images into the right classes. Semi-supervised GAN and our proposed model are both based on GANs. But the scenarios to be addressed by semi-supervised GANs and our proposed model are different. Semi-supervised GANs try to deal with the scenario when there are a small number of labeled samples but an amount of unlabeled samples. On the other hand, our proposed model tries to deal with the scenario when only a limited number of labeled samples are available. The conditional GANs [36] are proposed to introduce extra information to both generator and discriminator.

2.3 GAN-based Data Augmentation

Several GAN-based methods [54,12,47] have been proposed for data augmentation. Zhu *et al.* [54] proposed a GAN-empowered data augmentation framework to augment training data in emotion classification tasks. Wang *et al.* [47] combined GAN with several classic data augmentation methods to improve the performance for palmprint recognition. Frid-Adar *et al.* [12] used GAN to augment medial images for improved performance in liver lesion classification tasks. However, most existing GAN-based data augmentation models are limited to the binary classification, which can not be used for the multi-class classification (e.g., the Chinese character recognition task). In this paper, we propose a GAN-based deep data augmentation framework. By emphasizing the importance of label information in the discriminator of GAN, this framework effectively solves the confusion caused by too many categories in multi-classification.

3 Methodology

In this section, we first review GANs briefly in subsection 3.1. Secondly, GANs for the conditional image generation like Chinese characters are presented in subsection 3.2. Next, the deep data augmentation (DDA) framework is introduced in subsection 3.3. Finally, we present the details of the implementation architecture of DDA in subsection 3.4.

3.1 Generative Adversarial Networks

The GANs consist of a generator G and a discriminator D. The two roles are trained in a competitive way. G aims to estimate the data distribution p_{data} and generate indistinguishable samples from the estimated distribution p_g. On the other hand, the target of D is to distinguish whether an sample is from p_{data} or p_g. The learning process can be formalized as follows. A random vector z, which is the input for G, is drawn from a uniformly distribution p_{z}. Then z is mapped by the generator function $G(\cdot)$ to a generated sample $G(\text{z})$. Both the samples drawn from p_{data} and p_g are mapped by the discriminator function $D(\cdot)$ to output a scalar value $p(d = 1|\text{x})$, where $d = 1$ means that x is from p_{data}. During the learning process, D tries to output $p(d = 1|\text{x}; \text{x} \sim \text{p}_{\text{data}}) = 1$ and output $p(d = 1|\text{x}; \text{x} \sim \text{G}(\text{z})) = 0$. On the contrary, in terms of G, it tries to output $p(d = 1|\text{x}; \text{x} \sim \text{G}(\text{z})) = 1$. G and D are trained alternately and progress simultaneously. The two-stage objective of GANs can be defined as follows:

$$\begin{aligned} &\min_D V_1(D) = \mathbb{E}_{\text{x}\sim\text{p}_{\text{data}}(\text{x})}\big[-\log D(\text{x})\big] + \mathbb{E}_{\text{z}\sim\text{p}_{\text{z}}(\text{z})}\big[-\log(1 - \text{D}(\text{G}(\text{z})))\big], \\ &\min_G V_1(G) = \mathbb{E}_{\text{z}\sim\text{p}_{\text{z}}(\text{z})}\big[-\log D(G(\text{z}))\big]. \end{aligned} \tag{1}$$

3.2 GANs for the Conditional Image Generation

As stated before, we adopt conditional GANs for generating image generation, which is more suitable for domains like Chinese characters. The generator G and discriminator D are conditioned on label vectors y. For the application of Chinese characters recognition, the size of one-hot label vectors y is large. Conditioning on y directly will lead to the mode collapse problem when the size of y is much larger than the size of the noise vector z, because the label vectors y will dominate the input of the generator, which makes the generator difficult to encode the semantics to the noise vector z. Moreover, conditioning on y directly will also lead to the problems of infeasible memory cost if the size of y is sufficiently large. To solve this problem, we propose to map the large label vector y into a small vector $\Phi(\text{y})$ before conditioning. To further alleviate the mode collapse problem, we adopt least squares GAN [34] for our

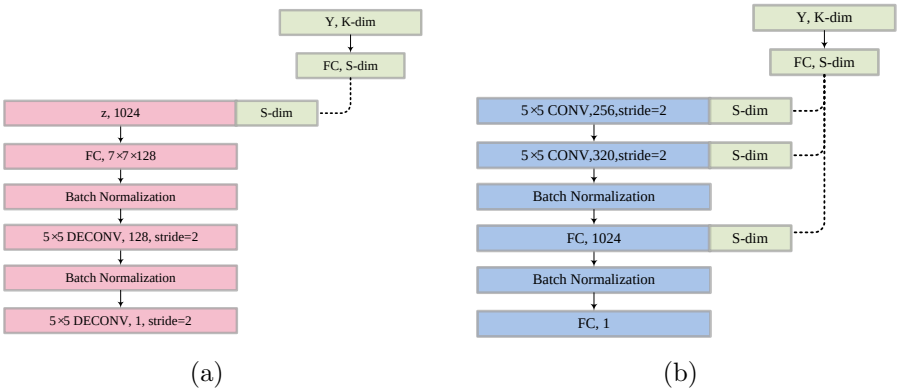

Fig. 3 Left (a): The generator. Right (b): The discriminator. FC represents the fully-connected layer, CONV represents the convolutional layer and DECONV represents the deconvolutional layer.

model, which has been proven more stable in practice. Thus the objective of GANs for the conditional image generation can be defined as follows:

$$\begin{aligned}\min_D V_2(D) &= \mathbb{E}_{\mathrm{x}\sim \mathrm{p_{data}(x)}}\left[\left(1 - D(\mathrm{x}|\Phi(\mathrm{y}))\right)^2\right] + \mathbb{E}_{\mathrm{z}\sim \mathrm{p_z(z)}}\left[\left(\mathrm{D(G(z)}|\Phi(\mathrm{y}))\right)^2\right],\\ \min_G V_2(G) &= \mathbb{E}_{\mathrm{z}\sim \mathrm{p_z(z)}}\left[\left(1 - D(G(\mathrm{z})|\Phi(\mathrm{y}))\right)^2\right].\end{aligned} \tag{2}$$

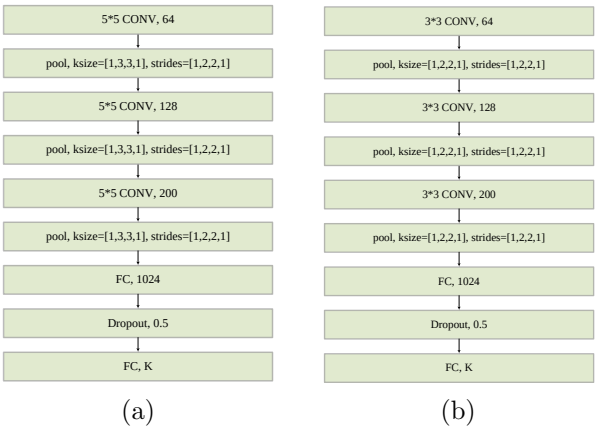

Fig. 4 Left (a): The architecture of the classifier is used for HWDB1.0. Right (b): The architecture of the classifier for MNIST, which is the same as baseline Adaptive-DA. FC represents the fully-connected layer and CONV represents the convolutional layer.

3.3 Deep Data Augmentation Framework

Let $\mathcal{X} = \{(\mathrm{x}^{(1)}, \mathrm{y}^{(1)}), (\mathrm{x}^{(2)}, \mathrm{y}^{(2)}), \ldots (\mathrm{x}^{(\mathrm{m})}, \mathrm{y}^{(\mathrm{m})})\}$ be a set of m labeled samples, where $\mathrm{y}^{(\mathrm{i})} \in \mathbb{R}^{\mathrm{K}}$ is the one-hot label vector and K is the number of classes. We define a classifier model trained on $\mathcal{X}$ as $p_c(\mathrm{y}|\mathrm{x}; \mathcal{X})$. The variance of p_c is critical to its performance on test set. To decrease the variance, one of useful methods is to collect more training samples. Data augmentation is one of the effective techniques to collect more training samples without human labelling. The traditional data augmentation techniques are almost based on human specific transformations, which are determined by empirically results. Unlike traditional data augmentation techniques, the proposed DDA framework is an end-to-end learning model. The DDA framework estimates the class-dependent data distribution $P(\mathrm{x}|\mathrm{y}; \mathcal{X})$ such that it is able to sample new data $\mathcal{X}'$ from the estimated data distribution. Given $\mathcal{X}$ and $\mathcal{X}'$, we are able to train a new classifier model $p_c(\mathrm{y}|\mathrm{x}; \mathcal{X}, \mathcal{X}')$ with lower variance.

The architecture of the proposed DDA framework is shown in Figure 2, which consists of three parts: a generator G, a discriminator D and a classifier C. G estimates the data distribution $p(\mathrm{x}|\mathrm{y}; \mathcal{X})$ and generates new data $\mathcal{X}'$. D identifies whether an image is from p_{data} or from p_g. C classifies which class an image x belongs to. All the three parts are differentiable neural networks and can be trained end-to-end. Particularly, a random input vector z is mapped by a mapping function $G(\cdot)$ to an generated image $\mathrm{I_g}$, i.e. $\mathrm{I_g} = \mathrm{G(z)}$. G is composed of several linear mapping layers and deconvolutional layers. In terms of D, images from p_{data} and p_g are both forwarded through the mapping function $D(\cdot)$ to output scalar values $D_{\mathrm{x}\sim\mathrm{p_{data}}}(\mathrm{x})$ and $D_{\mathrm{x}\sim\mathrm{p_g}}(\mathrm{x})$. On the other hand, images from p_{data} and p_g are assembled and forwarded through a mapping function $C(\cdot)$ to the label vectors $C(y_i = 1|\mathrm{x})$.

During the learning process, the parameters of G, D, and C are updated alternately. We begin to update the parameters of C when the learning step is larger than a threshold τ_c because of the poor quality of images generated by G at the beginning of the learning process. The threshold τ_c is determined empirically. In our experimental setting, we set τ_c to $5,000$ and $10,000$ for the tasks of digit recognition and Chinese character recognition, respectively. The training process of DDA is formally presented in Algorithm 1. After training, the learned classifier C which is trained on $\mathcal{X}$ and $\mathcal{X}'$ can be used alone to predict labels of new data.

3.4 Model Architecture for DDA

The network architectures of G and D are shown in Figure 3. The one-hot label vector y is mapped to a small vector $\Phi(\mathrm{y})$ by a fully-connected layer, and $\Phi(\mathrm{y})$ is concatenated to the noise vector. Then the joint representation is sent into a full connection layer to map to the distributed representation of features. Finally, two deconvolution layers generate the specific details of the picture according to the distributed representation, which can be considered

Algorithm 1 Training process of deep data augmentation using minibatch stochastic gradient descent.

for number of training iterations **do**

- Sample m random inputs $\{\mathrm{z}^{(1)}, \ldots, \mathrm{z}^{(\mathrm{m})}\}$ from uniform distribution p_z.
- Sample m real images $\{(\mathrm{x}^{(1)}, \mathrm{y}^{(1)}), \ldots, (\mathrm{x}^{(\mathrm{m})}, \mathrm{y}^{(\mathrm{m})})\}$ from data distribution p_{data}.
- Generate m fake images $\{(G(\mathrm{z}^{(1)}), \mathrm{y}^{(1)}), \ldots, (\mathrm{G}(\mathrm{z}^{(\mathrm{m})}), \mathrm{y}^{(\mathrm{m})})\}$ by mapping $\{(\mathrm{z}^{(1)}, \mathrm{y}^{(1)}), \ldots, (\mathrm{z}^{(\mathrm{m})}, \mathrm{y}^{(\mathrm{m})})\}$ through G.
- Update θ_d by descending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\left(1 - D(\mathrm{x}^{(\mathrm{i})} | \Phi(\mathrm{y}^{(\mathrm{i})})\right)^2 + \left(\mathrm{D}(\mathrm{G}(\mathrm{z}^{(\mathrm{i})}) | \Phi(\mathrm{y}^{(\mathrm{i})})\right)^2 \right].$$

- Update θ_g by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \left(1 - D(G(\mathrm{z}^{(\mathrm{i})}) | \Phi(\mathrm{y}^{(\mathrm{i})})\right)^2.$$

if #Iteration $> \tau_c$ **then**

- Assemble $\{(\mathrm{x}^{(1)}, \mathrm{y}^{(1)}), \ldots, (\mathrm{x}^{(\mathrm{m})}, \mathrm{y}^{(\mathrm{m})})\}$ and $\{(G(\mathrm{z}^{(1)}), \mathrm{y}^{(1)}), \ldots, (\mathrm{G}(\mathrm{z}^{(\mathrm{m})}), \mathrm{y}^{(\mathrm{m})})\}$ to form a minibatch of $2m$ images and we denote it as $\{(\mathrm{x}'^{(1)}, \mathrm{y}'^{(1)}), \ldots, (\mathrm{x}'^{(2\mathrm{m})}, \mathrm{y}'^{(2\mathrm{m})})\}$
- Update θ_c by descending its stochastic gradient:

$$\nabla_{\theta_c} \frac{1}{2m} \sum_{i=1}^{2m} \left(- \sum_{j=1}^{K} y_j'^{(i)} \log C(y_j'^{(i)} = 1 | \mathrm{x}'^{(\mathrm{i})}) \right).$$

end if

end for

as the inverse process of convolution. For the discriminator D, we first extract the contour information of the image by a convolution layer. Then further extract the distributed features of the image according to the contour features through the second convolution layer. Finally, the distributed features learned above are mapped to the sample marker space by a full connection layer, and the decision is made to get the result of image classification. It needs to be emphasized that, in order to prevent over-fitting or unstable training due to the over-complexity of the network, only two convolution layers are designed for the generator and discriminator. The activation layers are not shown in Figure 3. G adopts ReLU activation except for the last layer with sigmoid activation and D adopts LeakyReLU activation. Figure 4 shows the network architectures of C. In order to correspond to the classifier used on the baseline and highlight the effectiveness of GAN-based data enhancement, the simplest Lenet structure is used in classifier C. We compare the performance

between $p_c(\mathrm{y}|\mathrm{x};\mathcal{X},\mathcal{X}')$ and $p_c(\mathrm{y}|\mathrm{x};\mathcal{X})$ with the same network architecture to demonstrate the effectiveness of the proposed framework.

Note that $\Phi(\mathrm{y})$ is only concatenated to the first layer of the generator because the first layer decodes the high-level semantics and the class information is a high-level semantic. Furthermore, we empirically verify that concatenating $\Phi(\mathrm{y})$ to the first layer of the generator is enough to guide the generator to generate images in the corresponding class, and doing so reduces the computational and memory cost. On the other hand, $\Phi(\mathrm{y})$ is concatenated to three layers of the discriminator because we want the discriminator to pay more attention to the class information of the images rather than only whether the images are real or fake, and this is also empirically verified by experiments.

4 Experiments

In this section, we first introduce the details of experimental datasets in subsection 4.1. The experimental settings are discussed in subsection 4.2. The experimental results of the proposed DDA framework and other baseline models on Chinese character recognition and digit recognition are then discussed in subsection 4.3 and 4.4, respectively.

4.1 Datasets

We evaluate the proposed technique on four datasets. The details of the four datasets are presented in Table 1. The first one is HWDB1.0 [32] which is a handwritten Chinese character dataset. It contains $3,740$ classes and each category contains about 455 samples. We randomly select K classes from the training set to see how the accuracy improvements vary for a different number of classes, where K varies from 100 to $3,740$. The second one is MNIST [28] which is a handwritten digit dataset. We randomly select M samples from the training set to see how the accuracy improvements vary for a different number of samples, where M varies from 50 to $60,000$. The third one is EMNIST [6] which is an extended dataset of MNIST. The fourth one is Small-NORB [29] which contains 3D objects from 5 categories.

Table 1 Statistics of the datasets.

Dataset	#Classes	#Training	#Test
HWDB1.0	$3,740$	$1,246,991$	$309,684$
MNIST	10	$60,000$	$10,000$
EMNIST	47	$114,800$	$16,800$
Small-NORB	5	$24,300$	$24,300$

Table 2 Details of optimization methods.

Task	Model	Optimizer	Batchsize	LR	β
Chinese	G, D	Adam	32	2e-4	0.5
Chinese	C	Mementum	64	1e-3	-
Digit	G, D	Adam	64	2e-4	0.5
Digit	C	Adam	256	2e-4	0.9

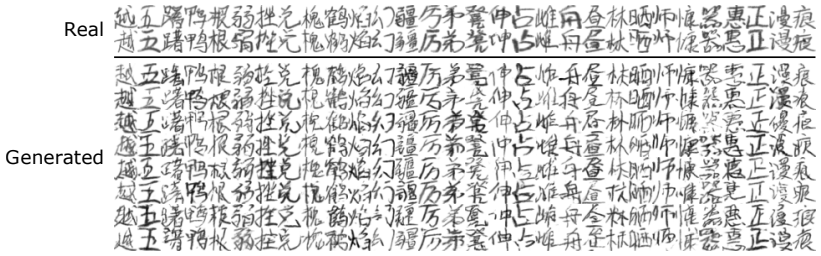

Fig. 5 Generated labeled Chinese character images. The top two lines are real images from training data. Images in the same column belong to the same category.

Table 3 Error rates(%) on HWDB1.0. Results are averaged over 10 times of training.

Model	Top-1 Error(%) Given K classes		
	100	1,000	3,740
CNN without DA	7.77 ± 0.54	11.56 ± 0.17	15.43 ± 0.17
CNN with traditional DA	7.12 ± 0.71	11.17 ± 0.19	15.13 ± 0.22
Ours	2.53 ± 0.19	7.89 ± 0.13	11.20 ± 0.28

4.2 Experimental Settings

Our proposed model is implemented using TensorFlow [1]. After experiments on the validation set, the details of hyperparameter settings are shown in Table 2. Note that we adopt Momentum [41] for optimizing the classifier C in the task of Chinese character recognition. The reason is that we observe momentum can converge quickly for Chinese character recognition while Adam can hardly converge.

4.3 Experimental Results on HWDB 1.0

We evaluate the proposed model on the HWDB1.0 dataset. We use the following two models as the baseline methods: 1) CNN models without data

Table 4 Error rates(%) on MNIST. Results are averaged over 10 times of training.

Model	Top-1 Error(%) Given M training samples		
	100	500	$1,000$
CNN without DA	22.13 ± 0.55	6.57 ± 1.05	4.05 ± 0.17
Ours	18.84 ± 0.56	5.71 ± 0.96	3.67 ± 0.18
	$5,000$	$10,000$	$60,000$
CNN without DA	1.76 ± 0.15	1.32 ± 0.09	0.75 ± 0.07
Ours	0.98 ± 0.15	0.91 ± 0.17	0.41 ± 0.04

augmentation; 2) CNN models with traditional manually designed data augmentation. The manually designed data augmentation strategies include skew, random noise, and affine transformation. We use the same network architecture for the classifier of all models. We show some randomly generated Chinese characters by our model in Figure 5, and we can see that the generated images are readable. Results are averaged over 10 times of training for each model. As Table 3 shows, our model reduces the error rates significantly for all tasks. Compared with the traditional data augmentation method, our model outperforms it by 4.59% and 3.93% for the tasks with 100 classes and 3,740 classes, respectively.

From Table 3, we also observe that our model is more effective for the task with 100 classes (7.77% $\rightarrow$ 2.53%) than the one with 3,740 classes (15.43% $\rightarrow$ 11.20%). The reason for this result is that the diversity of generated characters in each class decreases as the number of classes increases, which limits the effectiveness of our data augmentation strategy. Therefore, improving the diversity of generated images is critical to the performance of our model. In practice, one solution to this problem is to first train multiple conditional GANs, each of which contains a part of classes (e.g., 100 classes), and then assemble the results of these models. Using this method, the diversity of generated characters can be improved significantly, and the error rate for the task with 3,740 classes can be further reduced to 10.5%.

4.4 Experimental Results on MNIST

To further evaluate the effectiveness of our proposed model, we evaluate it on the MNIST dataset and compare it with two state-of-the-art data augmentation techniques. We evaluate the performance with a different number of images. The number of images M varies from 100 to $60,000$. We randomly select M images and the number of images from each class is balanced. Table 4 shows the results, and we have two major observations. First, the proposed

model is able to reduce the error rate for all the given M significantly. The smaller the M, the greater the effect of data enhancement. This shows that the GAN-based data enhancement can effectively alleviate the over-fitting problem by increasing the number of training samples while ensuring that the distribution of transformed samples is the same as the original data distribution. Second, the proposed model has a lower standard error, which indicates that it can reduce the model variance.

Table 5 Error Reduction on MNIST.

Model		Top-1 Error(%) Given M training samples	
		$5,000$	$60,000$
[24]	Without DA	-	1.61
	With DA	-	1.60
	Reduction	-	0.62%
Adaptive-DA[11]	Without DA	1.84	-
	With DA	1.03	-
	Reduction	44.02%	-
AlignMNIST[18]	Without DA	-	0.65
	With DA	-	0.44
	Reduction	-	32.30%
Ours	Without DA	1.76	0.75
	With DA	0.98	0.41
	Reduction	44.32%	45.33%

Comparison with State-of-the-art Methods. Adaptive-DA [11] and AlignMNIST [18] are used as the baseline methods, which are two state-of-the-art data augmentation techniques. We set the same classifier network architecture as Adaptive-DA to demonstrate the effectiveness of the proposed framework. We present the comparison results in Table 5, where the error rates of Adaptive-DA and AlignMNIST are reported in literatures [11] and [18], respectively. As the network architectures of baseline methods and our model are different, we compare the performance using the percentages of error reductions, rather than the absolute values of error rates. Table 5 shows that our proposed model outperforms Adaptive-DA and AlignMNIST considerably. Adaptive-DA achieves data augmentation by an automatic and adaptive algorithm for choosing the transformations of the samples. The disadvantages of the difference between the distribution of transformation samples and original data in Adaptive-DA leads to a negative impact on the performance. AlignM-

Table 6 Error rates(%) on EMNIST. Results are averaged over 5 times of training.

Model	Top-1 Error(%) Given M training samples per class			
	15	25	50	100
CNN without DA	26.06	21.65	18.49	16.22
DAGAN [2]	23.93	19.74	17.22	15.20
Ours	22.53	18.72	16.90	15.01

NIST uses an unsupervised approach to learn the image transformations which appear within each class. Although AlignMNIST can be trained end-to-end, a limitation is that it needs to align the observations in order and learn the transformations within the aligned observations. This limitation leads to a performance drop for AlignMNIST. The GAN-based data augmentation method proposed in this paper learns the data augmentation algorithms from the data automatically, it generates more samples for training according to data distribution, avoiding the defects in Adaptive-DA and AlignMNIST.

Comparison with Semi-supervised GAN [39]. Semi-supervised GAN and our proposed model are both based on GANs. The difference is that semi-supervised GAN is to deal with the scenario of training with a limited number of labeled samples but an amount of unlabeled samples, while our proposed model is to improve the performance when only limited labeled samples are given. From the reported results in literature [39], we can see that semi-supervised GAN performs better for small values of M. But when M is relatively large (e.g., $1,000$), semi-supervised GAN can hardly reduce the error rate. On the contrary, Table 4 shows that our proposed model is still able to reduce the error rate from $M = 1,000$ to $M = 60,000$.

Analysis of Overfitting. We plot the learning curves for $M = 100$ in Figure 6. The learning curve of CNN model indicates that it suffers from a significant overfitting problem. The reason is that the CNN model can overfit 100 samples very easily and find a "tricky" solution for these samples, but this solution performs very badly on the test set. On the contrary, the learning curve of our proposed model shows more stable, because the generated new images by our model can increase the variety of the training samples and make the CNN model difficult to overfit the training samples.

4.5 Experimental Results on EMNIST

We also conduct experiments on the EMNIST [6] dataset which is an extension of MNIST datasets. The EMNIST dataset contains 131,600 samples from 47 classes. For this experiment, we compare with a state-of-the-art method called DAGAN [2]. Following [2], we conduct the experiments when given

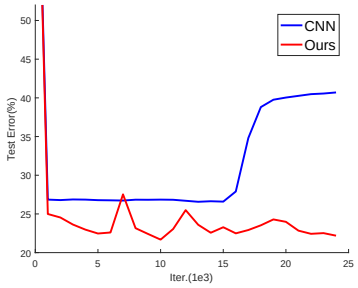

Fig. 6 Learning curves for $M = 100$.

Fig. 7 Examples from the Small-NORB dataset.

Table 7 Error rates(%) on Small-NORB.

Model	Top-1 Error(%)
CNN without DA	6.80
CNN with traditional DA	6.49
Adaptive-DA [11]	4.02
Ours	3.01

$\{15, 25, 50, 100\}$ training samples per class. Table 6 presents the results, and the results are averaged over 5 times of training. We can observe that our method can improve the accuracy of the CNN classifier by 3.53% when given 15 training samples per class. Compared with DAGAN, our method outperforms it for all experiments, especially for the experiment with 15 training samples per class.

4.6 Experimental Results on Small-NORB

We also conduct experiments on a more challenging dataset, Small-NORB [29]. The Small-NORB dataset contains 3D objects from 5 categories: four-legged animals, human figures, airplanes, trucks, and cars. Fig. 7 shows some examples from this dataset. For this experiment, we also compare with Adaptive-DA[11] and use the same classifier network architecture as Adaptive-DA for a fair comparison. The results are presented in Table 7. We can observe that both Adaptive-DA and our method can improve the performance of the classifier significantly and our method outperforms Adaptive-DA by more than 1%.

4.7 Discussion and Limitation

From the above experiments, we can conclude that our method can improve the performance of the classifier in general, especially when the number of training data is not sufficient. However, there are also some limitations of our method. The major limitation is that our data augmentation framework is based on GANs, thus if GANs cannot generate high-quality images for some complex datasets (e.g., scene datasets with many objects), the performance of our model will be limited. Another limitation is that the performance of our method depends on the diversity of the generated images. For some scenarios (e.g., datasets with a huge number of classes), the low diversity of the generated images limits the accuracy gain of our method. Therefore, improving the image quality and diversity of GANs is critical to the performance of our method.

5 Conclusion and Future Work

In this study, an end-to-end deep data augmentation framework based on generative adversarial networks is proposed. Specifically, we also describe the detailed methodology for DDA framework including the generative adversarial networks, GANs for the conditional image generation, the generic DDA framework, the training algorithm for DDA, and the detailed model architecture for implementing DDA framework. Different from traditional data augmentation methods that rely on human specification, the proposed DDA framework can learn the augmentation model from the data. The benefits of the data-driven model include reduced human work to design augmentation algorithms and higher performance improvements. We evaluate the proposed method on four real-life datasets. The experimental results demonstrate that the DDA outperforms baselines including one manually well-designed DA method and two state-of-the-art data augmentation methods.

One possible future work is to increase the diversity of generated images by introducing some priors to GANs. Increasing the diversity of the generated images can help the classifier to improve more. Another possible future work is to combine the proposed method and traditional augmentation methods. If the transformations learned from the data do not overlap traditional transformations, the combination of these transformations is likely to reach better performances.

Acknowledgement

The research of this work has been supported by the Dean's Research Fund 2018-19 (FLASS/DRF/IDS-3), Departmental Collaborative Research Fund 2019 (MIT/DCRF-R2/18-19) of The Education University of Hong Kong, and the Faculty Research Grant (DB21A9) of Lingnan University, Hong Kong.

References

1. Abadi, M., Agarwal, A., Barham, P., et al: TensorFlow: Large-scale machine learning on heterogeneous systems (2015)
2. Antoniou, A., Storkey, A., Edwards, H.: Augmenting image classifiers using data augmentation generative adversarial networks. In: International Conference on Artificial Neural Networks (2018)
3. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein gan. arXiv:1701.07875 (2017)
4. Brock, A., Donahue, J., Simonyan, K.: Large scale gan training for high fidelity natural image synthesis. arXiv:1809.11096 (2018)
5. Chen, Y., Li, W., Sakaridis, C., Dai, D., Van Gool, L.: Domain adaptive faster r-cnn for object detection in the wild. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
6. Cohen, G., Afshar, S., Tapson, J., van Schaik, A.: Emnist: an extension of mnist to handwritten letters. arXiv:1702.05373 (2017)
7. Cubuk, E.D., Zoph, B., Mané, D., Vasudevan, V., Le, Q.V.: Autoaugment: Learning augmentation policies from data. CoRR **abs/1805.09501** (2018). URL `http://arxiv.org/abs/1805.09501`
8. Cui, X., Goel, V., Kingsbury, B.: Data augmentation for deep neural network acoustic modeling. 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) pp. 5582–5586 (2014)
9. Denton, E., Chintala, S., Szlam, A., Fergus, R.: Deep generative image models using a laplacian pyramid of adversarial networks. In: Advances in Neural Information Processing Systems (NIPS), pp. 1486–1494 (2015)
10. Dixit, M., Kwitt, R., Niethammer, M., Vasconcelos, N.: Aga: Attribute-guided augmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7455–7463 (2017)
11. Fawzi, A., Samulowitz, H., Turaga, D., Frossard, P.: Adaptive data augmentation for image classification. In: IEEE International Conference on Image Processing (ICIP) (2016)
12. Frid-Adar, M., Diamant, I., Klang, E., Amitai, M., Goldberger, J., Greenspan, H.: Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification. Neurocomputing **321**, 321–331 (2018)
13. Girshick, R.: Fast R-CNN. In: International Conference on Computer Vision (ICCV) (2015)
14. Gong, Y., Wang, L., Guo, R., Lazebnik, S.: Multi-scale orderless pooling of deep convolutional activation features. In: ECCV, pp. 392–407 (2014)
15. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems (NIPS), pp. 2672–2680 (2014)
16. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of wasserstein gans. In: Advances in Neural Information Processing Systems, pp. 5767–5777 (2017)
17. Han Zhang Ian Goodfellow, D.M.A.O.: Self-attention generative adversarial networks. arXiv:1805.08318 (2018)
18. Hauberg, S., Freifeld, O., Larsen, A.B.L., III, J.W.F., Hansen, L.K.: Dreaming more data: Class-dependent distributions over diffeomorphisms for learned data augmentation. In: Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, AISTATS 2016, Cadiz, Spain, May 9-11, 2016, pp. 342–350 (2016)
19. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Computer Vision and Pattern Recognition (CVPR) (2016)
20. Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. Neural Networks **2**(5), 359 – 366 (1989)
21. Hu, W., Huang, Y., Wei, L., Zhang, F., Li, H.: Deep convolutional neural networks for hyperspectral image classification. Journal of Sensors **2015** (2015)
22. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Synthetic data and artificial neural networks for natural scene text recognition. In: Workshop on Deep Learning, NIPS (2014)

23. Jha, G., Cecotti, H.: Data augmentation for handwritten digit recognition using generative adversarial networks. Multimedia Tools and Applications **79**, 35055–35068 (2020)
24. Jorge, J., Vieco, J., Paredes, R., Sanchez, J.A., Benedi, J.M.: Empirical evaluation of variational autoencoders for data augmentation. In: VISIGRAPP (2018)
25. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. arXiv:1710.10196 (2017)
26. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. arXiv:1812.04948 (2018)
27. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems (NIPS), pp. 1097–1105 (2012)
28. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. In: Proceedings of the IEEE, pp. 2278–2324 (1998)
29. LeCun, Y., Huang, F.J., Bottou, L.: Learning methods for generic object recognition with invariance to pose and lighting. In: Computer Vision and Pattern Recognition (CVPR) (2014)
30. Li, W., Chen, C., Zhang, M., Li, H., Du, Q.: Data augmentation for hyperspectral image classification with deep cnn. IEEE Geoscience and Remote Sensing Letters **16**(4), 593–597 (2019)
31. Li, Z., Guo, J., Jiao, W., Xu, P., Liu, B., Zhao, X.: Random linear interpolation data augmentation for person re-identification. Multimedia Tools and Applications (2018). DOI 10.1007/s11042-018-7071-5. URL `https://doi.org/10.1007/s11042-018-7071-5`
32. Liu, C.L., Yin, F., Wang, Q.F., Wang, D.H.: Icdar 2011 chinese handwriting recognition competition. In: Proceedings of the 2011 International Conference on Document Analysis and Recognition, ICDAR '11, pp. 1464–1469 (2011)
33. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional models for semantic segmentation. In: Computer Vision and Pattern Recognition (CVPR) (2015)
34. Mao, X., Li, Q., Xie, H., Lau, R.Y., Wang, Z., Smolley, S.P.: Least squares generative adversarial networks. arXiv:1611.04076 (2016)
35. Mariani, G., Scheidegger, F., Istrate, R., Bekas, C., Malossi, C.: Bagan: Data augmentation with balancing gan. arXiv:1803.09655 (2018)
36. Mirza, M., Osindero, S.: Conditional Generative Adversarial Nets. arXiv:1411.1784 (2014)
37. Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral normalization for generative adversarial networks. arXiv:1802.05957 (2018)
38. Nowozin, S., Cseke, B., Tomioka, R.: f-gan: Training generative neural samplers using variational divergence minimization. arXiv:1606.00709 (2016)
39. Odena, A.: Semi-supervised learning with generative adversarial networks. arXiv:1606.01583 (2016)
40. Paulin, M., Revaud, J., Harchaoui, Z., Perronnin, F., Schmid, C.: Transformation pursuit for image classification. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3646–3653 (2014)
41. Qian, N.: On the momentum term in gradient descent learning algorithms. Neural Networks **12**(1), 145 – 151 (1999)
42. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. In: International Conference on Learning Representations (ICLR) (2015)
43. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X., Chen, X.: Improved techniques for training gans. In: Advances in Neural Information Processing Systems (NIPS), pp. 2226–2234 (2016)
44. Simard, P.Y., Steinkraus, D., Platt, J.C.: Best practices for convolutional neural networks applied to visual document analysis. In: Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR) - Volume 2, pp. 958– (2003)
45. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556 (2014)
46. Springenberg, J.T.: Unsupervised and Semi-supervised Learning with Categorical Generative Adversarial Networks. arXiv:1511.06390 (2015)

47. Wang, G., Kang, W., Wu, Q., Wang, Z., Gao, J.: Generative adversarial network (gan) based data augmentation for palmprint recognition. In: 2018 Digital Image Computing: Techniques and Applications (DICTA), pp. 1–7. IEEE (2018)
48. Wang, J., Perez, L.: The effectiveness of data augmentation in image classification using deep learning. Convolutional Neural Networks Vis. Recognit (2017)
49. Wang, S.H., Sun, J., Phillips, P., Zhao, G., Zhang, Y.D.: Polarimetric synthetic aperture radar image segmentation by convolutional neural network using graphical processing units. Journal of Real-Time Image Processing **15**(3), 631–642 (2018)
50. Yin, X., Yu, X., Sohn, K., Liu, X., Chandraker, M.: Feature transfer learning for deep face recognition with long-tail data. CoRR **abs/1803.09014** (2018). URL `http://arxiv.org/abs/1803.09014`
51. Yu, Q., Lam, W.: Data augmentation based on adversarial autoencoder handling imbalance for learning to rank (2019)
52. Zeng, S., Zhang, B., Gou, J.: Learning double weights via data augmentation for robust sparse and collaborative representation-based classification. Multimedia Tools and Applications **79**, 20617–20638 (2020)
53. Zhong, Z., Zheng, L., Kang, G., Li, S., Yang, Y.: Random erasing data augmentation. CoRR **abs/1708.04896** (2017). URL `http://arxiv.org/abs/1708.04896`
54. Zhu, X., Liu, Y., Qin, Z., Li, J.: Data augmentation in emotion classification using generative adversarial networks. arXiv preprint arXiv:1711.00648 (2017)