IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 18 January 2023, accepted 6 February 2023, date of publication 8 February 2023, date of current version 14 February 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3243473

RESEARCH ARTICLE

DPBA-WGAN: A Vector-Valued Differential Private Bilateral Alternative Scheme on WGAN for Image Generation

DANHUA WU[1], **WENYONG ZHANG**[1], **AND PANFENG ZHANG**[1,2]

[1]Department of Information Science and Technology, Guilin University of Technology, Guilin 541006, China
[2]Guangxi Key Laboratory of Embedded Technology and Intelligent System, Guilin University of Technology, Guilin 541006, China

Corresponding author: Panfeng Zhang (panf_zhang@glut.edu.cn)

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61862019, in part by the Guangxi Science and Technology Base and Talent Special Project Guike AD19110067, in part by the Scientific Research Foundation of Guilin University of Technology under Grant GLUTQD2017065, and in part by the Guangxi Key Laboratory Fund of Embedded Technology and Intelligent System.

ABSTRACT The large amount of sensitive personal information used in deep learning models has attracted considerable attention for privacy security. Sensitive data may be memorialized or encoded into the parameters or the generation of the Wasserstein Generative Adversarial Networks (WGAN), which can be prevented by implementing privacy-preserving algorithms during the parameter training process. Meanwhile, the model is also expected to obtain effective generated results. We propose a vector-valued differential private bilateral alternative (DPBA) algorithm, a novel perturbation method for the training process. The vector-valued Gaussian (VVG) noise involving functional structure information is injected into the WGAN to generate data with privacy protection, and the model is verified to satisfy differential privacy. The bilateral alternative noise can eventually randomly perturb the gradient and generates informative feature-rich samples. The dynamic noise and vector-based perturbation approach ensure privacy strength. After extensive evaluation, our algorithm outperformed state-of-the-art techniques in terms of usability metrics for all validation datasets. The downstream classification accuracy for the generated Mnist was 97.04%, whereas that for the Fashion-Mnist dataset was 80.91%. Mnist improved the average accuracy of the neural network classifier by at least 16.81%, and Fashion-Mnist by at least 3.55%. In the multichannel generation tasks, the binary classification accuracy improved by at least 10.4% compared to CelebA, and the accuracy of the Street View House Numbers SVHN was as high as 86.1%. The perturbation method proved highly resilient to gradient attack recovery under simulated gradient attacks.

INDEX TERMS Data generation, deep learning model, differential privacy, noisy perturbation, WGAN.

I. INTRODUCTION

With in-depth research and rapid development of artificial intelligence (AI) technology in recent years, AI applications have penetrated all aspects of industrial production and human life. AI technology represented by deep learning generally needs to achieve a better model effect through the analysis and training of a significant number of labeled samples. However, these samples typically contain sensitive information regarding an entity. A significant amount of sensitive information about the user was incorporated into the final training results after convolution and pooling layers. Existing attack methods against deep learning models can extract private information through certain means [1], which eventually leads to privacy leakage.

Various protection strategies based on differential privacy (DP) have been proposed to address these issues: Noise perturbation in a classifier based on the nearest neighbor algorithm [2] provides strict privacy protection during the data analysis. Combining with conditional filtering of noise

The associate editor coordinating the review of this manuscript and approving it for publication was Diana Gratiela Berbecaru.

based on an adaptive Gaussian mechanism [3] to prevent excessive noise, achieving the expected utility and privacy. Using the p-Power exponential mechanism(EM) [4] when the noise variance is quite a small relative to the signal and the dimension is not too high. Variational Bayesian privacy-preserving frameworks based on the optimal Bayesian inference method [5] are another way to solve the high cumulative privacy loss caused by noise. Training with DP models based on decision trees [6] to calculate the attribute weights, which can influence the degree of the noise and reduces the negative impact of the DP on data usability. Differential privacy has recently emerged as an accepted standard for defending against possible privacy threats in federated learning [7], [8]. Whether it is a client-server architecture that relies on a central server [9], [10] or an end-to-end network architecture [11], it is necessary to provide dual protection for internal training and external model publishing. The above shows the importance and application prospect of privacy protection, DP remains a research hotspot in academia for privacy conservation in machine learning, and its outcomes continue to be demonstrated in practical applications [12], [14].

Distinguishing it from the general direct application of DP techniques to the original data features [15], [16] or the original data release, the generative adversarial network (GAN) with DP algorithms implanted in it protects private data during the generation training process [17], [20]. Then, instead of publishing authentic data, data platforms publish desensitized data with utility, thereby reducing the probability of privacy disclosure. No doubt that the traditional DP framework [21] limits the expressiveness of the data after injecting noise. It is definitely significant that the GAN framework when embedding DP, generated data that can be useful for a series of subsequent assignments and tasks without compromising personal privacy. During the processing of all records of a dataset by a GAN, the DP requires recording the contribution of each component to the total privacy budget and then adding a random perturbation that is scaled appropriately. Our study focused on the relationship between privacy and usability of the generation. To improve the usability of the generated data with a privacy guarantee, we propose a WGAN based on a vector-valued differential private bilateral alternative (DPBA) scheme for privacy data generation. We impose a dynamic Gaussian noise in vector form, denoted as vector valued Gaussian (VVG), on the vector-valued cost according to the indices. Even if the shape of this Gaussian distribution is reconstructed, our method obtains a fixed privacy budget. VVG is a special form of tensor-valued Gaussian (TVG) [22] that can yield high-quality data with tighter noise bounds. Dynamically varying Gaussian noise was used to render the model gradient parameters more resistant to attack. A variant of $dyn\,[S, \sigma]$ [23] was used in the algorithm to cause independently and identically distributed Gaussian noise to vary randomly within a certain range during each training round. The training method of the GAN was coupled with the Wasserstein distance with a gradient penalty to facilitate control of the gradient flow. Considering the introduction of the gradient penalty, the vector-valued cost function is divided into two parts, both of which require the imposition of a dynamically varying VVG to ensure global DP. The main contributions of our study are as follows:

1. Dynamic noise by vector form injection into the vector-valued cost reduces the relative privacy loss while satisfying the higher availability of the generated data. The training procedure does not require modification of the internal structure of the network, which can improve training efficiency.

2. The innovative division of the cost vector into two parts, in which the two VVG mechanisms control the corresponding cost vectors, yields a balance between privacy preservation and usability.

The remainder is as follows: Section II presents related work and the motivation for our work. Section III introduces the related knowledge. Section IV describes the algorithm details of the proposed DPBA approach and presents the analysis and proof. Section V discusses and compares the results of the study. Finally, Section VI concludes the research.

II. RELATED WORKS

Before discussing our work, it is necessary to provide a detailed analysis of the theory, and the advantages or the disadvantages of existing representative DP algorithms on generative adversarial networks.

To preserve the privacy information of the training process, DP-GAN [24]was the first to propose a moderate noise addition on the gradient to satisfy the privacy of generative adversarial networks. Lorenzo et al [25] proposed a clipping-attenuation strategy to form a noise addition. Differentially Private Conditional GAN (DP-CGAN) [26] proposed a conditional model to generate corresponding labels and data to protect privacy. It adds noise to the gradient after using different clipping thresholds according to the generator and discriminator loss paradigms. Private GAN (Pri-GAN) [18] proposed an optimized discriminator with gradient estimation for the determination of clipping-bound C and then added noise to the gradient. These privacy-preserving training methods are based on DP stochastic gradient descent (DP-SGD), where the gradients are clipped according to the ℓ_2 parametric and preset boundaries when updating the weights of each layer of the network. The refreshed gradients were aggregated, and a Gaussian mechanism was attached in each round. However, the direct perturbation of the gradients makes it more difficult for the global parameters to converge, and the usability of this approach is often worse. Gradient Sanitized WGAN (GS-WGAN) [27] improves the basic DP-SGD by adjusting the loss function to alleviate the need for gradient clipping. Similarly to the Private Aggregation of Teacher Ensembles (PATE-GAN) [28], it deploys multiple discriminator networks trained on different parts of the dataset, amplifying privacy through subsampling. As a result, this approach must consider the sharing of parameters across multiple discriminators, the desires for equipment and arithmetic can also be extremely significant. In contrast to the gradient-plus-noise method, in particular, to improve the availability of generated data, we adopted a complementary noisy cost approach based on distance transmission perturbation. It avoids the problem of adding repeated noise and makes the operation faster and more convenient. The interactive training approach ensures that the noise factor is propagated in a circular chain to achieve the same privacy effect as the SGD perturbation.

Considering the random characteristic mean embedding of the data distribution, the DP Mean Embeddings with Random Features (DP-MERF) [29] use the optimal transmission method of the Maximum Mean Discrepancy (MMD) estimator. For adding noise to the estimator to achieve data desensitization, DP-MERF was computed using a reciprocal evaluation of the kernel function using points extracted from the real and generated data distributions. Nonetheless, compared with the MMD matrix-associated loss function, the objective function based on the Wasserstein distance is more general for high-dimensional data. Sinkhorn [30] introduced a semi-biased loss based on the optimal transmission algorithm of Sinkhorn divergence. Splitting the batch of generated data, calculating cross-term (biased) and self-term (debiased) losses by Sinkhorn iterations, and then adding noise to the biased losses to protect privacy. The Sinkhorn distance optimization algorithm obviously increases the network computational complexity while calculating the loss function. The proposal of WGAN [31], [32] promoted the development of DP-GAN, an optimization-based transmission algorithm, but its essence is objective function perturbation. In our study, a similar scheme is used, but the difference is that the noise is made more complex and the optimization cost is divided into two parts. The noise was added first, and then calculated the vector-valued cost. The noise also contributed to the optimization. The bilateral alternative effect is demonstrated by the fact that when one part of the loss is perturbed, the noise-added part is used as a perturbation term to satisfy the DP while optimizing the parameters through the other part, and vice versa. It is also necessary to pay attention to the effectiveness of adding noise when perturbing the objective function. For example, differentially private GAN [33] adds noise directly to the discriminator loss, and the noise factor as a constant term in reverse derivation has no variations. Thus, the privacy of each parameter cannot be guaranteed.

III. PRELIMINARIES

This section briefly describes the preliminary knowledge necessary for the DPBA-WGAN, including an overview of Renyi differential privacy (RDP) and generative adversarial networks. Our main purpose is to use this relaxed version of DP such that the training of WGAN satisfies both differential privacy and better image generation quality.

A. RENYI DIFFERENTIAL PRIVACY

Differential privacy was first proposed by Dwork for the privacy leakage problem of databases, and is considered to have strict DP [34].

Definition 1: A randomized mechanism M gives ε-differential privacy if for all datasets x and x' differ on at most one element, and all $S \subseteq Range\,(M)$.

$$P_r[M(x) \in S] \leq e^{\varepsilon} \times P_r[M(x') \in S] \tag{1}$$

where P_r denotes the probability. The definition depends on two adjacent datasets x and $x' \in X$, M provide privacy protection by randomization of the output results.

The relaxed version of (ε, δ)-Differential privacy [35], [36] introduces a δ factor to the original concept of DP.

Definition 2: A randomized mechanism $M : x \to R$ with domain x and range R satisfies (ε, δ)-differential privacy if for any two adjacent inputs $x, x' \in X$and for any subset of outputs $S \subseteq Range\,(M)$ it holds that inequation,

$$P_r[M(x) \in S] \leq e^{\varepsilon} \times P_r[M(x') \in S] + \delta \tag{2}$$

δ is a relaxation term for accepting DP to a certain point of dissatisfaction, denoting that it is ε-DP except with probability δ. Although different from the definition of equation (1), when $\delta = 0$, it is essentially pure DP.

Achieving DP usually requires the addition of controllable noise to reduce the sensitivity of query results and to make the budget ε smaller and the privacy protection effect higher. The Gaussian mechanism is a prototypal (ε, δ) −differentially private algorithm that allows the release of an approximate answer to an arbitrary query with values in R^n. The mechanism is defined as follows:

$$G_\sigma M_f(x) \triangleq M_f(x) + \mathcal{N}(\mu, \Delta_2 f^2 \sigma^2) \tag{3}$$

where G_σ is the gaussian mechanism, $\mathcal{N}$ denotes Gaussian distribution with mean μ and standard deviation $\Delta_2 f \sigma$, the definition for ℓ_2-sensitivity $\Delta_2 f$ [36] is as follows:

$$\Delta_2 f \triangleq \max_{x,x'} \left\| M_f(x) - M_f(x') \right\|_2 \tag{4}$$

RDP is a broader definition extended [37] by the above, sharing many properties through modifications that make differential privacy a helpful and general tool, making gaussian mechanisms more versatile and simple.

Definition 3 (Renyi Divergence): For the expectation $\mathbb{E}_x$ of the two probability distributions P and Q defined across R, the Renyi divergence of order $\alpha > 1$ is:

$$D_\alpha\,(P||Q) \triangleq \frac{1}{\alpha - 1} \log \mathbb{E}_{x\sim Q} \left(P\,(x) / Q\,(x)\right)^{\alpha} \tag{5}$$

Definition 4 ((α,)-RDP): A randomization mechanism $f : x \to R$ is said to have Renyi differential privacy of order α, or referred to as $(\alpha,)$-RDP, if for any adjacent dataset x, $x' \in S$, there exists

$$D_\alpha \left(f\,(x)\,||f\,(x')\right) \leq \varepsilon \tag{6}$$

Proposition 1 (From RDP to $(,\delta)$-DP): If f denotes an $(\alpha,)$-RDP mechanism, then it also satisfies $\left(\varepsilon + \left(\log\left(1/\delta\right)\right)/(\alpha-1), \delta\right)$-differential privacy for any $0 < \delta < 1$.

Corollary 1: The algorithmic mechanism f is a combination of n ε-differential privacy mechanisms, and let $0 < \delta < 1$ in such a way that $\log\left(1/\delta\right) \geq \varepsilon^2 n$. Then we have f satisfying $\left(\varepsilon', \delta\right)$-differential privacy, where:

$$\varepsilon' \triangleq 4\varepsilon\sqrt{2n\log\left(1/\delta\right)} \tag{7}$$

In fact, $(\alpha,)$-RDP can also be expressed as $(\varepsilon_\delta, \delta)$-DP for any given probability, where $0 < \delta < 1$. The above definition and proposition also inherit the properties of ε-DP. The privacy guarantee in the ε-DP type is defined as $e^{-\varepsilon} \times P_r[M(x') \in S] \leq P_r[M(x) \in S] \leq e^{\varepsilon} \times P_r[M(x') \in S]$, while the privacy protection in $(\alpha,)$-RDP is defined as $\left\{e^{-\varepsilon} \times P_r[M(x') \in S]\right\}^{\frac{\alpha}{\alpha-1}} \leq P_r[M(x) \in S] \leq \left\{e^{\varepsilon} \times P_r[M(x') \in S]\right\}^{\frac{\alpha-1}{\alpha}}$.

This allows intuitive and quantitative privacy budget concepts to be combined with advanced composition theorems. When k random functions $(M_1, M_2, \cdots, M_k)$ act on the same dataset, they are called a composition, written as $M_{1:k}$. Assuming that the number of iterations is T and each random function M_i satisfies $(\varepsilon_i, \delta_i)$-DP, the composition formed by the series of T Gaussian mechanisms M_i fulfills (ε, δ)-DP.

B. GAN ARCHITECTURE

Generative adversarial networks [38] are a framework for estimating generative models using an adversarial process proposed in 2014 by Ian J. Goodfellow et al. It consists of a generator (G) and a discriminator (D), where G captures the underlying distribution of real data samples and generates new data samples, and D is a binary classifier that discriminates whether the input is real or generated. Both the generator and discriminator can be deep neural networks. The proposal of GAN started a technical revolution, but training instability is a common problem, WGAN [39] concluded that Wasserstein distance is the most suitable for GAN training after analyzing Kullback-Leibler (KL) divergence, Jensen–Shannon (JS) divergence, Total Variation (TV) distance, and Wasserstein (W) distance. Since when the distribution of functions in space satisfies the K-Lipschitz condition, where K is the factor, the W distance is everywhere continuous and almost everywhere differentiable for the joint distribution represented by the predicted and labeled distributions. The objective function of D is:

$$\begin{aligned}\mathcal{L}_D = \mathbb{E}_{\tilde{x}\sim\mathbb{P}_g}\left[D\left(\tilde{x}\right)\right] - \mathbb{E}_{x\sim\mathbb{P}_r}\left[D\left(x\right)\right] \\ + \mathbb{E}_{\hat{x}\sim\mathbb{P}_{\hat{x}}}\left[\left(\left\|\nabla_{\hat{x}} D\left(\hat{x}\right)\right\|_2 - \mathrm{K}\right)^2\right]\end{aligned} \tag{8}$$

The objective function of G is as follows:

$$\mathcal{L}_G = -\mathbb{E}_{\tilde{x}\sim\mathbb{P}_g}\left[D\left(\tilde{x}\right)\right] \tag{9}$$

where $\tilde{x}$ is the data generated by the G, this data and the real data x are used for the training of the D. The gradient distribution during training obeys K-Lipschitz. When training the generator, feedback from the discriminator is needed. D aims to identify as much fake data as possible, while G generates as much data as possible that can deceive D. The two gambling with each other in this way.

GANs have formidable expressive capabilities to perform arithmetic operations in the latent vector space, and convert them into computations in the corresponding feature space. Info-GAN [40], GAN based on U-net [41], gaussian mixture model (GMM) [42], two-stage GAN [43], etc. are all variants of competitive training approaches by generative adversarial networks.

IV. METHOD IMPLEMENTATION

In this section, we describe the design of the WGAN framework based on DPBA, elaborate on the design of the algorithm, and provide further proof and privacy analysis of this schema. The purpose of our method is to prevent attackers from recovering original datasets that contained sensitive information from gradients or parameters, thereby reducing the possibility of data leakage containing sensitive information during the training of deep learning models. Preventing the generator from generating samples with private data or reflecting critical features, maintaining the generation with better usability. Finally, we conclude that the proposed method can achieve this goal.

A. DP WGAN SCHEMA

In practical instances, the distribution of the original training data is unknown and must be inferred by empirical estimation. The requirement to be implemented here is to synthesize the simulated sample estimates directly using the gaming process of the WGAN mechanism, release and input them into the actual downstream network. The risk of privacy leakage of the original data can be limited during the training and release of the data. With a suitable training scheme, it is possible to resist model attacks and improve the privacy protection level.

Our design to achieve the above goal is to add differentially private variables to the traditional training intermediate process, over the gradient or loss function, to constrain the trained model to obtain some private information extracted by the attacker. We propose a new strategic target for the loss, as shown in FIGURE 1, where the vector-valued loss with a sensitive stream is split into two portions. When one portion of the vector is perturbed, the same portion of the other vector is undisturbed, culminating in the use of a new cost for the discriminator update and generator training. The essence of the WGAN is that the sample generation task is a game of G and D. The discriminator needs to identify whether the input data are real or fake, whereas the generator continuously generates fake samples based on noisy feedback. After repeated iterations, the final expectation was reached with an approximate distribution of the original data. Because the quality of the generation depends on the performance of the discriminator, the original data features should be

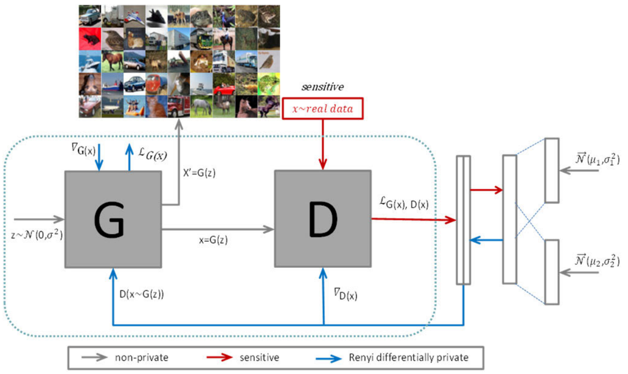

sensitive
x~real data
$\nabla_{G(x)}$
$\mathcal{L}_{G(x)}$
X'=G(z)
$z \sim \mathcal{N}(0,\sigma^2)$
G
x=G(z)
D
$\mathcal{L}_{G(x),\,D(x)}$
$\vec{\mathcal{N}}(\mu_1,\sigma_1^2)$
$\vec{\mathcal{N}}(\mu_2,\sigma_2^2)$
D(x~G(z))
$\nabla_{D(x)}$
non-private
sensitive
Renyi differentially private

FIGURE 1. Overview of the DPBA-WGAN structure.

shadowed. To achieve the WGAN with the DPBA algorithm, we decided to train the discriminator first, and the sensitive parameters can flow into the generator during the training process. According to the flow of information in FIGURE 1, the design will be friendlier to the downstream generator by desensitizing the discriminator parameters. In the update process of the generator's parameters θ_G, as shown in equation (10):

$$\theta_G^{(t+1)} := \theta_G^t - \eta \cdot \nabla_{(D(G(z));\theta)} \mathcal{J}\left(\mathcal{L}_{(D(x),G(z);\theta)}\right) \tag{10}$$

The loss $\mathcal{L}$ involving the parameters θ of discriminator D(x) and G(z), $\mathcal{J}$ is the Jacobian, $\nabla_{(D(G(z));\theta)}$ denotes the gradients when updating the parameters of the generator.

The backpropagation update requires the use of discriminator parameters. Although the generator is not exposed to the real data stream during training, the discriminator will carry the reflection and shadowing of sensitive statistics. Applying the noise mechanism to the discriminator vector-valued cost function, in this case, can effectively impose a privacy barrier.

Unlike prior work on adding noise to discriminator gradients, we focus on the interaction between the generator and discriminator because the network is more vulnerable to infection, and the adversary has more accessible information. Adding noise to the cost function is more straightforward and easy [33], and it is easier to operate without directly exchanging the parameters of the discriminator-generator interaction or modifying the gradient update process. From [33], we consider that the noise on the cost function cannot be un-functional during the parameters optimization. Thus, the noise term is added as a common factor of cost so that the noise can always contribute during backpropagation. The designated noise consists of two bilateral alternative counterparts, which are more privacy guaranteed and helpful for convergence. The change in the gradients during the experiment must be within the bounding range and usually requires clipping the target value. Nevertheless, our framework is based on a WGAN with a gradient penalty, in which the gradients satisfy the Lipschitz condition, resulting in a bounded gradient without additional processing of the target function or gradient.

B. DPBA-WGAN ALGORITHM

The specific implementation of DPBA-WGAN for image generation is described in Algorithm 1. Even though the generator does not directly access to sensitive training data, the mapped features can be inferred from the discriminator parameters. Thus, our algorithm preserves the parameters or raw data in the discriminator from being divulged to the generator, thereby making the generator as free of sensitive parameters as possible. Consequently, when the published model is physically attacked during an interaction, it is almost impossible for aggressors to obtain valuable information.

Algorithm 1 DPBA WGAN Training Process

```
Input: real dataset $X$, discriminator $D(\theta_D)$, generator $G(\theta_G)$, batch size B, Gaussian noise scale $\sigma_1, \sigma_2$, clipping coefficient $\Delta$, total epochs T, gradient penalty weights $\lambda$, learning rate $\alpha$, default hyperparameters $\beta_1, \beta_2$ etc.
Output: Differentially private generator G with parameters $\theta_G$
set the dataset X into subsets $\{X_l\}_{l=1}^{L}, L = \lceil len(X)/B \rceil$
for epoch t in range (0, T) do
    for $l$ in range (0, L) do
        Initialize discriminator $\theta_D$
        # here write the subset $X_l$ as $x$
        sample real data $\{x_i\}_{i=1}^{B} \sim x$
        sample latent vectors $\{z_i\}_{i=1}^{B} \sim z \leftarrow G(z)$
        a random number $\mu \sim U[0, 1]$
        $\hat{x} \leftarrow \mu x + (1-\mu) \cdot z$
        $grad_{\hat{x}} \leftarrow \frac{1}{B}\sum_{i=1}^{B} clip(grad_{\hat{x}}, \Delta) + \mathcal{N}(\mu_1, \sigma_{1_i}^2)$
        $\mathcal{L}_{\theta_D}(x, z) \leftarrow \frac{1}{B}\sum_{i=1}^{B}[\mathcal{L}_{\theta_D}(x_i) - \mathcal{L}_{\theta_D}(z_i)] \cdot N(\mu_2, \sigma_{2_i}^2)$
        $grad_{\theta_D} \leftarrow \nabla_{\theta_D}((\mathcal{L}_{\theta_D}(x, z) + \lambda grad_{\hat{x}}), \theta_D, \alpha, \beta_1, \beta_2)$
        update $\theta_D$ with differentially private $grad_{\theta_D}$
        Initialize generator $\theta_G$
        sample generated data $\{z_i\}_{i=1}^{B} \sim z \leftarrow G(z)$
        $grad_{\theta_G} \leftarrow -\nabla_{\theta_G}(\mathcal{L}_{\theta_D}(z), \theta_G, \alpha, \beta_1, \beta_2)$
        update $\theta_G$ with $grad_{\theta_G}$
    End for
End for
Return generator $G(\cdots, \theta_G)$
```

As shown in Algorithm 1, the entire dataset X is firstly divided into subsets according to batchsize B. For each training round of WGAN, the following steps are performed for each batch of sampled data:

The discriminator and generator are trained sequentially, and the discriminator D is trained first. Initialize the parameters θ_D of the discriminator, subsample a batch of real data X_l, and generate a batch of random noise data z_i using generator G.

The interpolation methodology was deployed to mix the positive and negative data, and the batch of data was treated as an arbitrary point in the entire subsampled space. The result is used as the input of the discriminator to find the gradient penalty item parameterized by the batch of $\hat{x}$. Adding the noise vector, i.e., VVG noise, to perturb the item. The upper bound of the penalty is consequently constrained to be Δ.

The real data x_i and the generated data z_i were used as the input of the discriminator, and calculated the vectored scores of each genuine or fictitious batch separately. Another VVG noise perturbation was applied to the score to disturb the Wasserstein distance $\mathcal{L}_{\theta_D}(x_i) - \mathcal{L}_{\theta_D}(z_i)$ of the real and fake data, which can also be used to judge the degree of model convergence.

After perturbation, the Wasserstein distance and penalty term $grad_{\hat{x}}$ are redefined as a cost in the form of vector valued (VV) in the ratio of $1 : \lambda$. The Adam optimizer is used to find the gradient of the discriminator and update the weights, whereby the perturbed updating process of each parameter of the discriminator is filtered out of the sensitive information.

Because the discriminator parameter θ_D provides original data privacy protection, the generator G is trained by simply regenerating a batch of fake data z. Based on the generator's cost obtained by the feedback of discriminator D, computing the gradient and updating the θ of the generator.

The algorithm ends up with a WGAN that satisfies Renyi differential privacy by applying VVG noise. This is because the discriminator parameters before updating the generator parameters can guarantee privacy, and the generator can still guarantee privacy. Therefore, a WGAN consisting of a discriminator trained first and a generator that needs to be updated can also protect privacy. This generative adversarial network is well protected in terms of privacy for both the training process and generated data after it is released.

C. SENSITIVITY

Considering a univariate statistic for a sample set in the privacy release problem, such as differential privacy, our method defines the maximum difference between the statistics $\mathcal{L}(X)$ and $\mathcal{L}(X')$ of two neighboring sample sets as sensitivity [44]. Then, adding vector-valued Gaussian noise to this statistic $\mathcal{L}$ is intended to make it impossible to determine whether it is computed from X or X', which makes it impossible for an attacker to determine which dataset of X and X' comes from, even after deriving the statistic. We characterized the sensitivity of $\mathcal{L}$ as follows:

$$\Delta S = \max_{X,X'} \left|\mathcal{L}(X) - \mathcal{L}(X')\right| \tag{11}$$

Here, the max symbol covers all adjacent sets, and the maximum value of the change in the output result for any single change in the distribution is noted as the sensitivity. We assume that the batchsize of the WGAN process is B. In accordance with [21], we can get the privacy budget by the sampling rate, training iterations, and sensitivity et al, where the sensitivity is need to be calculated as follows. Denote the Wasserstein distance or penalty term in Algorithm 1 as$\mathcal{L}_i(X)$, we theoretically clip this cost function so that it satisfies $\|\mathcal{L}_i(X)\| \leq C$, and add a Gaussian mechanism to this $\mathcal{L}_i(X)$ to make it satisfies Renyi difference privacy. Same principle as gradient descent to obtain sensitivity [45], the cost function can denote as $\mathcal{L}(X) = \frac{1}{|B|}\sum_{i=1}^{|B|} \mathcal{L}_i(X)$, we can get $\Delta S = 2C \big/ |B|$ in result, where C is the clipping threshold to bound $\|\mathcal{L}_i\|$. The gradient $grad_\theta$ of WGAN in the backpropagation will also impose a noise factor, and the gradient of the parameter update process is set in to satisfy 1-lipschitz continuity, $C_\theta = 1$. The gradient of the perturbation can also get the relative upper boundary C, so the above form of ΔS can naturally be obtained.

D. NOISE OPTIONS

Traditional DP mechanisms based on Laplace or Gaussian noise are tailored for scalar-valued query functions. Considering that the cost function can be represented in vector form, we redefined the cost value shape precisely and used a DP scheme for the vector-valued query function. The mechanism of the VVG algorithm is shown in Algorithm 2.

Algorithm 2 VVG Mechanism With Bilateral Alternative Noise

```
Input: $f(x) \in \mathbb{R}^{m \times 1}, \mu_1, \mu_2, \sigma_1, \sigma_2, \sigma_3, \sigma_4$
for i = 1, 2, ..., m:
    set choice = 0 or 1
    $\mu = \mu_1, \sigma_{\mu_1} = [\sigma_1, \sigma_2]$ ← choice value is 0
    $\mu = \mu_2, \sigma_{\mu_2} = [\sigma_3, \sigma_4]$ ← choice value is 1
    get random $i^{\text{th}}$ direction's variance $\sigma_i = \left[\sigma_{\mu_1}, \sigma_{\mu_2}\right]$
$\sum = (\sigma_1, \cdots, \sigma_i, \cdots, \sigma_{\text{m}})^{\text{T}}$
vector valued noise $z$ from $VVG_{(\text{m},1)}\left(\mu, \sum, I\right)$
Output: $f(x) + z$
```

The cost function is redefined as a vector in the batchsize dimension, projected to the batchsize dimensional component element, and compounded with noise in the batchsize dimension. From which, we get final noisy objective function. The objective cost in Algorithm 1 consists of two parts: the training vector-valued loss and the vectored penalty in the WGAN. Thus, the VVG perturbation applying to the vector valued function also has two parts, that is, $f(x) = (\mathcal{L}_1 + z_1) \circ (\mathcal{L}_2 + z_2)$. In our experiment, inspired by the dynamic noise concept of [23], the dynamic noise scale was obtained by setting the noise variance threshold to vary within a certain interval. To prevent increasing epsilon computational confusion or damaging the model by adding repeated noise, we use the bilateral alternative (BA) strategy. It is set that if the i^{th} dimensional value in z_1 matches VVG, the corresponding i^{th} value in z_2 is set to 0, and vice versa.

E. PRIVACY ANALYSIS AND PROOF

1) ANALYSIS OF PRIVACY WGAN

For better understanding the above algorithm, we discuss how the DPBA-WGAN can protect the privacy from the original to the generated data throughout the generative adversarial training process.

The WGAN training focuses on the privacy barrier of the discriminator. From the direction of the Renyi privacy flows (Section IV-A), it is known that the generator can naturally meet the privacy requirements after privacy upstream. To achieve differential privacy in deep learning network models, including WGAN, many existing studies are in the stochastic gradient descent (SGD) process, adding a sufficient amount of noise to implement. Usually, the large-scale noise added to the cost function at the time of generation cannot achieve privacy protection of network. Since gradient updating still use the sensitive parameters in the backpropagation process. In general, we need to define a function as the cost after performing the forward propagation algorithm for neural networks, measuring the error between the calculated output of the generation and the real score of the input subsamples. It is assumed that the input un-noised sample is x with loss $\mathcal{L}$. By back propagation, $w^{[l]}$ is denoted as the l^{th} layer parameter of the L-layer network, and the resulting $\partial\mathcal{L}/\mathcal{L}\partial w^{[l]}$ has coefficients x. The input value of x contains no additional noise, so the data privacy cannot be guaranteed. We implemented the WGAN by adding VVG noise in the calculation of the cost function to protect privacy while training and producing privacy samples. The network following these prerequisites while training:

1. Train the discriminator first, and each round of training should contain the same batch size of real and fake data as the input.

2. Use the bilateral alternative perturbation, ensure that the trained discriminator network no longer carries or reflects sensitive information in preparation for the training of the generator.

3. Combine the discriminators and generators of the above conditions into one whole, i.e., the WGAN deep neural network model.

4. The generator in the WGAN model generates batches of data and feeds them into the discriminator

5. According to the cost of desensitization, the noisy gradients are calculated of the WGAN, and update parameters with privacy guarantee.

The sensitive x may be encoded or reflected in the discriminator's parameters and gradients during training. The generator's parameters and gradients are computed based on the discriminator's parameters. Correlating the two can lead to some sensitive and informative messages being transferred to the generator's gradients and parameters. However, applying our DPBA method in the deep WGAN model can prevent x from contributing explicitly to the gradients, desensitize the flowing parameters, and generate insensitive data while protecting privacy from disclosure.

2) RDP PROOF OF THE WHOLE NETWOK

To better illustrate the flow of sensitive messages in the entire network, a simple network model, as shown in FIGURE 2, is illustrated as an example to show how the discriminator vector-valued cost with RDP affects the gradients during the parametric update when generating fake samples as the input in the WGAN structure.

Suppose the input data are x. This network consists of an m-layer generator network and an n-layer discriminator network, $w^{[l]}$, $b^{[l]}$ are the parameters of the l^{th} layer, with $z^{[l]}$ and $a^{[l]}$ caching intermediate values, and g is the activation function, then we have $z^{[l]} = w^{[l]}a^{[l]} + b^{[l]}, a^{[l]} = g\left(z^{[l]}\right)$, $z^{[1]} = w^{[1]}x + b^{[1]}$. Forward propagation calculates the loss value by $\mathcal{L}$. Since the noise is added in the bilateral alternative way, it can be seen as that the total noise adding to the $\mathcal{L}$ is obey Gaussian distribution. The addition of the above noise scheme, denoted as $\mathcal{N}$, allows the discriminator to satisfy RDP, where the privacy budget is $\varepsilon = r\left(\alpha^2\Delta S^2\right)/\left[2\sigma^2(\alpha-1)\right]$. After T iterations of RDP component theory, the final privacy budget is $\varepsilon' = \left(r\alpha^2\Delta S^2\right)/\left[2(\alpha-1)\sigma^2\right]\sqrt{2T\log(1/\delta)}$. The universal proof of RDP is shown in Appendix A.

The cost function is used for backward propagation to continuously update the parameters, assuming that vector perturbation has been performed on the VV cost function of each batchsize data. If we want to obtain the parameter $w_1^{[l]}$, then we add up the multiplications obtained from the derivation of the corresponding chain equations as follows:

$$
\begin{aligned}
\frac{\partial\mathcal{L}}{\partial w_1^{[1]}} = (&\frac{\partial\mathcal{L}}{\partial a'^{[n]}}\frac{\partial a'^{[n]}}{\partial z'^{[n]}}\frac{\partial z'^{[n]}}{\partial w'^{[n]}}\frac{\partial w'^{[n]}}{\partial a_1'^{[n-1]}}\frac{\partial a_1'^{[n-1]}}{\partial z_1'^{[n-1]}}\frac{\partial z_1'^{[n-1]}}{\partial w_1'^{[n-1]}} \\
&\cdots\frac{\partial a_1^{[2]}}{\partial z_1^{[2]}}\frac{\partial z_1^{[2]}}{\partial w_1^{[2]}}\frac{\partial w_1^{2}}{\partial a_1^{[1]}}\frac{\partial a_1^{[1]}}{\partial z_1^{[1]}}\frac{\partial z_1^{[1]}}{\partial w_1^{[1]}} \\
&+\frac{\partial\mathcal{L}}{\partial a'^{[n]}}\frac{\partial a'^{[n]}}{\partial z'^{[n]}}\frac{\partial z'^{[n]}}{\partial w'^{[n]}}\frac{\partial w'^{[n]}}{\partial a_2'^{[n-1]}}\frac{\partial a_2'^{[n-1]}}{\partial z_2'^{[n-1]}}\frac{\partial z_2'^{[n-1]}}{\partial w_2'^{[n-1]}} \\
&\cdots\frac{\partial a_1^{[2]}}{\partial z_1^{[2]}}\frac{\partial z_1^{[2]}}{\partial w_1^{[2]}}\frac{\partial w_1^{2}}{\partial a_1^{[1]}}\frac{\partial a_1^{[1]}}{\partial z_1^{[1]}}\frac{\partial z_1^{[1]}}{\partial w_1^{[1]}}+\cdots)\cdot\mathcal{N} \quad (12)
\end{aligned}
$$

Here, one of the derivative chains is chosen to specify how RDP budget is affecting the gradient update, as is shown in the chain corresponding to the bolded red line in FIGURE 2:

$$
\begin{aligned}
&\frac{\partial\mathcal{L}}{\partial a'^{[n]}}\frac{\partial a'^{[n]}}{\partial z'^{[n]}}\frac{\partial z'^{[n]}}{\partial w'^{[n]}}\frac{\partial w'^{[n]}}{\partial a_3'^{[n-1]}}\frac{\partial a_3'^{[n-1]}}{\partial z_3'^{[n-1]}}\frac{\partial z_3'^{[n-1]}}{\partial w_3'^{[n-1]}} \\
&\qquad\cdots\frac{\partial a_2^{[2]}}{\partial z_2^{[2]}}\frac{\partial z_2^{[2]}}{\partial w_2^{[2]}}\frac{\partial w_2^{2}}{\partial a_1^{[1]}}\frac{\partial a_1^{[1]}}{\partial z_1^{[1]}}\frac{\partial z_1^{[1]}}{\partial w_1^{[1]}}\cdot\mathcal{N} \quad (13)
\end{aligned}
$$

If only the generator is released, it is clear that $\partial z_1^{[1]}\big/\partial w_1^{[1]} = x$, where x is the coefficient of the aforementioned parameter w gradient. The input x is the data derived from the generator, not the real data, and does not contain sensitive information. Thus, we firstly described the discriminator, whose parameters are obtained from the gradient with real data x and perturbed when the noisy factor $\mathcal{N}$ is added. Combined with the fact that $\mathcal{L}$ satisfies Renyi differential privacy, it can be inferred that $\frac{\partial\mathcal{L}}{\partial a'^{[n]}}$ satisfies Renyi differential privacy [37], $\partial a'^{[n]}/\partial z'^{[n]}$, $\partial a_3'^{[n-1]}\big/\partial z_3'^{[n-1]}$, $\cdots$, $\partial a_1^{[1]}\big/\partial z_1^{[1]}$are also satisfied, and so do $\partial z'^{[n]}/\partial w'^{[n]}$, $\partial z_3'^{[n-1]}\big/\partial w_3'^{[n-1]}$, $\cdots$, $\partial z_1^{[1]}\big/\partial w_1^{[1]}$, and equation (13) satisfies Renyi differential privacy. Similarly, the derivative chains in equation (12) satisfies Renyi differential privacy as well, so $\partial\mathcal{L}\big/\partial w_1^{[1]}$ satisfies Renyi differential privacy. According to the inference, the gradients corresponding to the parameters $w^{[l]}$ and $b^{[l]}$ in the WGAN also satisfy the Renyi differential privacy.

At this point, it can be concluded that under the premise of the generated data and the real data distribution as the input of the training scheme, adding a suitable VVG noise perturbation to the discriminator VV cost before training the generator can ensure that the entire network update process satisfies Renyi differential privacy. The coefficients involved in the backpropagation process are perturbed that do not expose original data privacy. According to the chain transferability, the discriminator parameters sanitize sensitive information, and privacy in the backpropagation of the entire process of RDP will be guaranteed. Unlike the traditional cost function with noise, the noise size does not change with the gradient update, but the overall gradient is heading toward the optimal value. Thus, the noise perturbation method is not directly involved in progress, but functions as part of the optimization function and does not diminish with the decay of the gradient. When the optimal gradient is obtained with noise as a whole, that the model training is completely perfect. During each training round, the vital information of raw input data x may be leaked, and a hacker may extract the sensitive information in the model based on x as well as the weights in back-propagation. However, the gradients are affected by noise perturbations that satisfy differential privacy. The gradients with x are perturbed, and the input x is subjected to a fake distribution for generator. Therefore, intruders have a significant obstacle in separating sensitive information from the feedback of queries in the training response.

As can be seen above, the entire update process of the network satisfies the Renyi differential privacy, and the whole WGAN architecture satisfies Renyi differential privacy during the training process. Therefore, the fake generation by such a generator is satisfactory for privacy guarantee. We can conclude that applying VVG noise to the cost of a VV form can effectively help desensitize the data in WGAN deep training.

V. EVALUATION

This section describes the implementation and validation of the scheme using several datasets and evaluation metrics. We compared the experimental results with other methods, observed the effect of some hyper-parameter variations on the realized results, and verified the scheme's resistance against gradient attacks on a simple classification network accompanied by a presentation of the experimental procedure and results.

A. EVALUATION INSTRUCTIONS

1) DATASETS

Our algorithm was evaluated on image data using two typical datasets, Mnist [46] and Fashion-Mnist [47], with 60,000 training images coupled with 10,000 test images, with every image size of 28 × 28 and with a single channel. The SVHN dataset [48] consists of Google Street View House Numbers, which includes 73,257 training image samples. The test set consists of 26,032 test image samples, and the whole dataset consists of 10 categories, i.e., the numbers 0-9. CelebA [49] is an open dataset from the Chinese University of Hong Kong (CUHK), it has 202,599 images of 10,177 celebrity identities, and they are all well-labeled with features, which is a very promising dataset for face-related training. For the Mnist and Fashion-Mnist, suppose we know that a certain digit or cloth belongs to a certain person, when the attacker recovers the recurring data and compares it with the person, then we can determine whether the person is in the set. For SVHN, it's possible to sure whether one's house number is in the set. Or to get the pupil distance of a person's face from the recovered CelebA image, etc. More details of the datasets are in Appendix B.

2) EVALUATION METRICS

The purpose of the algorithm is to protect privacy while ensuring the high usability of the model. Privacy-preserving ability depends on the privacy budget ε. The smaller the value of ε, the better the privacy performance [21]. Two metrics were used in the experiments to verify the algorithm availability: the Frechet Inception Distance (FID) [50] was used to calculate the distribution gap between the generative data and authentic data, which can evaluate the quality of the generated privacy data. The privacy data released by the generator were used as the training set for the downstream classifier to evaluate the accuracy of the real test dataset. For comparison, the classification algorithm uses Logistic Regression (LR), Multilayer Perceptron (MLP), and Convolutional Neural Networks (CNN) [30]. Consistently, the parameters of the train and test batchsize is 256, the max iterations is 500. Using Adam optimizer with no weight decay, the learning rate is 0.001, and the betas is 0.999. For the classification training, 10 percent of the training set was used as the holding part for the early stop [51] mechanism, and the training was stopped if the accuracy of the validation set did not improve for 10 continuous rounds. When performing the downstream tasks, we additionally used the random_forest, Gaussian_nb, Bernoulli_nb, linear_svc, decision_tree, lda, adaboost, bagging, gbm, and xgboost classifiers, the detailed results are presented in Appendix C.

3) IMPLEMENTATION

To improve the quality of the generated data as much as possible and to ensure the robustness of the model, the experimental model refers to WGAN [39] and introduces a post-perturbation penalty term as well as the Wasserstein distance to ensure that the model satisfies differential privacy and generates privacy-preserving data. To elaborate on the generalizability of the algorithm, the Mnist and Fashion-Mnist datasets were initially preprocessed so that each label was trained independently, and both were experimented with using a convolutional neural network architecture.

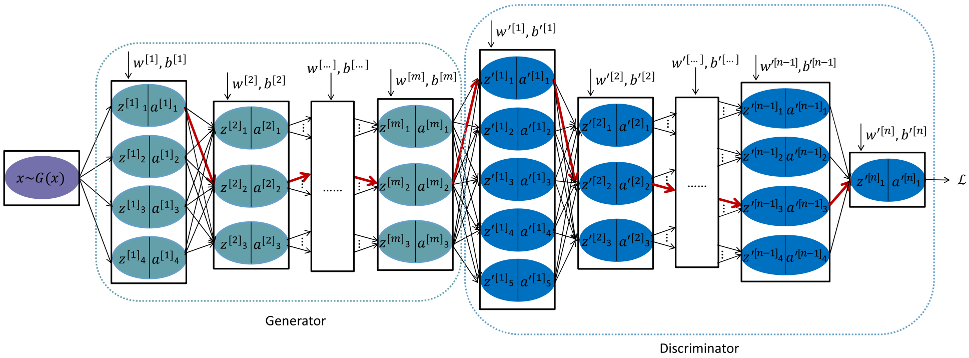

$x \sim G(x)$
$w^{[1]}, b^{[1]}$
$z^{[1]}{}_1$ $a^{[1]}{}_1$
$z^{[1]}{}_2$ $a^{[1]}{}_2$
$z^{[1]}{}_3$ $a^{[1]}{}_3$
$z^{[1]}{}_4$ $a^{[1]}{}_4$
$w^{[2]}, b^{[2]}$
$z^{[2]}{}_1$ $a^{[2]}{}_1$
$z^{[2]}{}_2$ $a^{[2]}{}_2$
$z^{[2]}{}_3$ $a^{[2]}{}_3$
$w^{[\dots]}, b^{[\dots]}$
......
$w^{[m]}, b^{[m]}$
$z^{[m]}{}_1$ $a^{[m]}{}_1$
$z^{[m]}{}_2$ $a^{[m]}{}_2$
$z^{[m]}{}_3$ $a^{[m]}{}_3$
Generator
$w'^{[1]}, b'^{[1]}$
$z'^{[1]}{}_1$ $a'^{[1]}{}_1$
$z'^{[1]}{}_2$ $a'^{[1]}{}_2$
$z'^{[1]}{}_3$ $a'^{[1]}{}_3$
$z'^{[1]}{}_4$ $a'^{[1]}{}_4$
$z'^{[1]}{}_5$ $a'^{[1]}{}_5$
$w'^{[2]}, b'^{[2]}$
$z'^{[2]}{}_1$ $a'^{[2]}{}_1$
$z'^{[2]}{}_2$ $a'^{[2]}{}_2$
$z'^{[2]}{}_3$ $a'^{[2]}{}_3$
$w'^{[\dots]}, b'^{[\dots]}$
......
$w'^{[n-1]}, b'^{[n-1]}$
$z'^{[n-1]}{}_1$ $a'^{[n-1]}{}_1$
$z'^{[n-1]}{}_2$ $a'^{[n-1]}{}_2$
$z'^{[n-1]}{}_3$ $a'^{[n-1]}{}_3$
$z'^{[n-1]}{}_4$ $a'^{[n-1]}{}_4$
$w'^{[n]}, b'^{[n]}$
$z'^{[n]}{}_1$ $a'^{[n]}{}_1$
$\mathcal{L}$
Discriminator

FIGURE 2. An example of parameter flows in WGAN Structure.

TABLE 1. Comparison results on Mnist and Fashion-Mnist($\delta = 10^{-5}$).

		DP-ε	FID↓	acc↑(%)				
				logistic_reg	mlp	cnn	average	calibrated
MNIST	Real	∞	1.02	92.56	97.58	98.85	96.33	100.00
	DP-GAN	10	354.68	48.39	57.40	69.31	58.37	60.59
	GS-WGAN	10	61.34	79.00	79.00	80.00	79.33	82.36
	DP-Sinkhorn	10	23.66	72.47	76.28	88.30	79.02	82.03
	DP-MERF	1	351.27	79.59	76.51	75.74	77.28	80.22
	Ours	1	4.47	87.58	93.38	97.04	92.67	96.20
Fashion-MNIST	Real	∞	1.49	84.41	87.79	90.99	87.73	100.00
	DP-GAN	10	370.77	59.03	57.84	64.40	60.42	68.87
	GS-WGAN	10	131.34	68.00	65.00	64.00	65.67	74.85
	DP-Sinkhorn	10	29.58	73.11	75.29	76.57	74.99	85.48
	DP-MERF	1	309.55	66.75	66.08	64.68	65.84	75.04
	Ours	1	28.17	74.77	77.26	80.91	77.65	88.51

The CNN-based generator uses 1 Linear layer and 3 convolutional layers, with Rectified Linear Unit (ReLU) activation function after each hidden layer and Tanh activation function in the output layer. The discriminator used two convolutional layers and two fully connected layers with the LeakyReLU activation function after each hidden layer. The CelebA dataset was trained based on a Progan [52] structure. SVHN was trained using the labeling approach of DCGAN [53], and also using the CNN-based discriminator and generator network, but with additional deeper layers to ensure the usability of the output. The algorithms designed for our experiment were executed using the four aforementioned datasets.

B. RESULTS AND COMPARISON

To verify the merits of our DPBA algorithm, the experimental results on Mnist and Fashion-Mnist are compared with the following state-of-the-art techniques available: DP-GAN [24], GS-WGAN [27], DP-Sinkhorn [30], and DP-MERF [29], as shown in TABLE 1.

In order to make the comparison results more reliable, all δ is set to 10^{-5}, the epsilon for DP-GAN, GS-WGAN, and DP-Sinkhorn is 10. The epsilon values for DP-MERF and our scheme can be smaller, set to 1 depending on the impact of the noise scale on the final outcomes. The evaluation results of every methodology were materialized on 6 K generated samples. DP-GAN and DP-MERF directly used the source code provided by [29], and the results of GS-WGAN were derived from [27], [30]. While the DP-Sinkhorn is reproduced using the Sinkhorn optimal distance transmission algorithm proposed by the authors, combined with the CNN framework in our experiment, the parameter m is set to 1.

FIGURE 3 provides the visualization samples of the generation, which roughly show the generated quality of each approach. The quantitative results in TABLE 1 show that the FID metrics of our scheme exceed all benchmarks. For example, the FID value on Mnist improves 77.6 times against DP-MERF (4.47 vs. 351.27) and approximately 10 times higher on Fashion-Mnist (28.17 vs. 309.55). Despite a larger privacy budget, its FID score still outperforms other algorithms, such as DP-Sinkhorn. Our proposed method

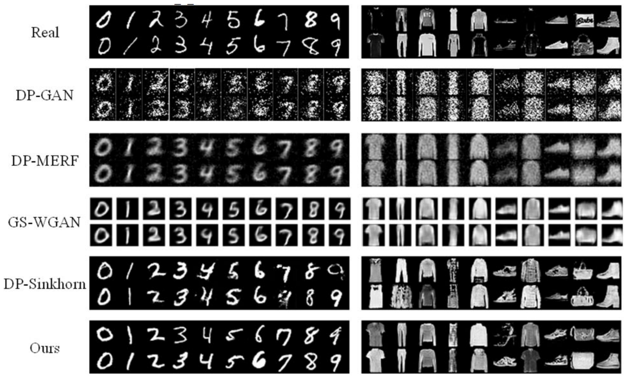


FIGURE 3. Generated samples on Mnist and Fashion-Mnist via various methods.

generates samples that capture the statistical properties of the original data better, thus contributing to the performance of downstream tasks. As can be seen in the table, the metrics of cnn, mlp, and logistic_regretion outperform the classification results of other schemes on both datasets, with an average accuracy of 92.67% for Mnist, whereas the best baseline (GS-WGAN) was only 79.33%. The average accuracy of Fashion-Mnist was 77.65%, which is a relative improvement of 3.55% over the best baseline (DP-Sinkhorn). In conclusion, our scheme has significantly improved various metrics and the quality of generated images with smaller privacy overhead. More details of the classification results can be found in Appendix C.

C. INFLUENCIES OF HYPERPARAMETERS

Privacy and availability can be influenced by many non-aligned hyperparameters such as iteration rounds, subsampling rate, and noise scale. In FIGURE 4, it can be seen that the changing of a certain hyperparametric value will have an impact on the output results.

As a whole, as the privacy budget ε increases, the privacy can be less protected, but the values of accuracy and FID increase, meaning the usability of the model becomes higher. The first row of each figure indicates the effect of the parameter variables on the accuracy of the downstream classification results, and the FID shows the trend of the output statistical quality of feature information benefiting for the usage. FIGURE 4(a) shows that the classification accuracy of the generated privacy data improves by 10.9%-41.2% over the comparison line when changing the number of iterations, and the value of FID has a 5 to 87.5 times improvement. When trained using our scheme, the model can converge faster with a smaller privacy budget. When the value of epsilon is less than two, the model can generate higher-quality image, which improves the training efficiency of the model and enhances the output effectiveness of the network. The efficiency and quality of the DP-GAN, DP-Sinkhorn, and DP-MERF algorithms were all inferior to our solution. As shown in FIGURE 4(c), both the DP-MERF and our models maintain robustness when a larger noise scale is used. Setting the privacy loss to less than one, as the noise scale increases, the generated samples still maintain a significant advantage in the CNN and MLP testing tasks. In the comparison of FID, it can be seen that, with the same privacy protection budget, our scheme produces images of much better usable information than DP-MERF, with a more accurate grasp of high-dimensional pixel features. In the evaluation of the sampling rate, it is generally illustrated that the larger the sampling rate, the higher the generative quality, and the better the privacy guarantee and usability. The best privacy results and image quality for a sampling rate of 1/600 in FIGURE 4(b) are also consistent with the derivation of the privacy overhead in Section IV (equation (14)). More details on the experimental results are provided in Appendix C.

TABLE 2. Comparison results on CelebA.

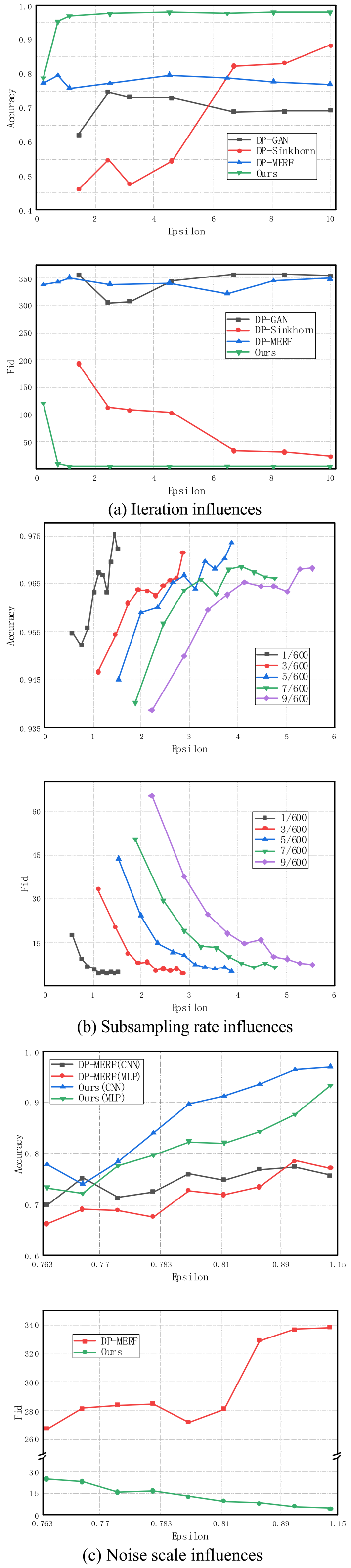
	(ε,δ)-DP	Fid↓	acc↑(%)	
			cnn	mlp
Real	∞	2.79	98.58	97.31
Datalens	10,10-5	320.80	-	72.90
DP-Sinkhorn	10,10-6	168.40	76.20	75.80
Ours	10,10-6	31.35	86.64	83.68

D. EXPERIMENT ON RGB IMAGE

We further evaluated the privacy and usability of our algorithm on RGB images. The privacy model was first trained on the CelebA dataset with the same privacy loss ε and δ as DP-Sinkhorn for comparison purposes, which ensured privacy magnitude consistency.

The targets of the downstream output are all for the binary classification task of identifying gender, with 0 for male and 1 for female, and FIGURE 5 shows the visualization of the generated samples. By naked-eye observation, the Datalens method cannot identify gender at all, and DP-Sinkhorn can faintly tell the outline of gender. In contrast, our RGB results contain richer figurative information, and we can directly determine gender through observation at a glance. TABLE 2 presents a quantitative comparative analysis. In terms of FID and classification accuracy, our method achieves the best performance among the three, and the generative image quality (FID) is improved by 4.36 times than DP-Sinkhorn, and the accuracy is also improved by more than 10% in the dichotomous classification test. The comparative evaluation results show that our schema is not only able to synthesize RGB images that provide useful information for downstream classification, but also that the model can better serve the downstream task under the same privacy conditions. Thus, we achieve better privacy and high usability of the model.

As a more complex dataset than MNIST and Fashion-MNIST, SVHN also achieved good generation effects with a privacy budget of 10. The FID distance between the real training and test datasets was calculated to be 1.02. FIGURE 6(a) shows some samples of the real dataset, and FIGURE 6(b) shows the visualization results of the generated dataset



(a) Iteration influences

(b) Subsampling rate influences

(c) Noise scale influences

FIGURE 4. Analysis of hyperparameters on Mnist($\delta = 10^{-5}$).

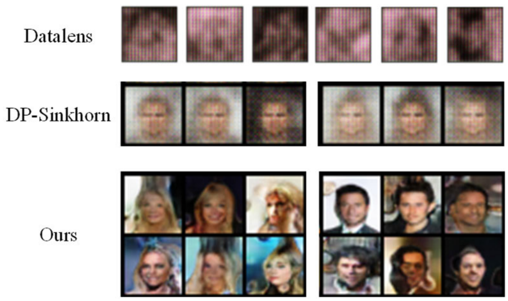
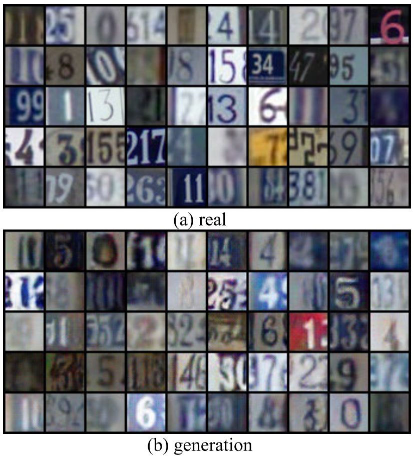


FIGURE 5. Generated samples on CelebA.



(a) real

(b) generation

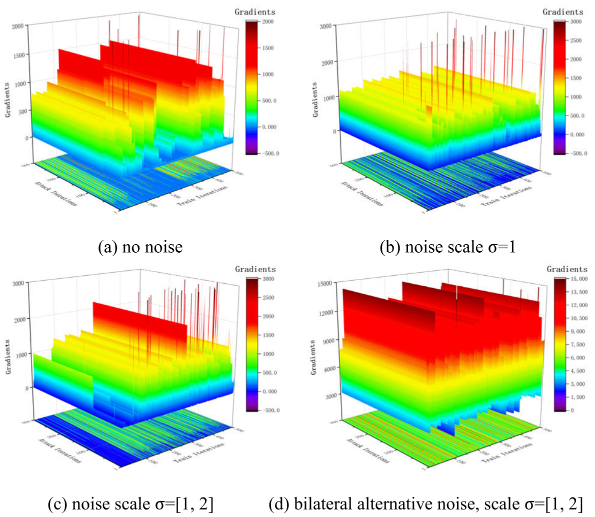
FIGURE 6. Samples of SVHN.

from the differential-privacy WGAN with an FID indication of 268.58. The produced dataset was tested with a CNN-based downstream classifier, and the final accuracy was 86.1%, which is a good trade-off between privacy and usability for our model compared with the accuracy of 90.02% for the real training dataset.

E. RESISTENCE OF GRADIENTS ATTACK

A simple classification network was customized to test the ability of our DPBA perturbation strategy against gradient attacks. We performed a simulation of a gradient attack on a single image (digit 7) during the training of the Mnist dataset. Privacy tactics consist of (i)no privacy, (ii)fixed noise scale, (iii)dynamic interval noise scale, and (iv)bilateral alternative dynamic interval noise. We can obtain gradient fitting results, which show the success and failure of the aggression. The gradient leakage resistance of each perturbation decision was evaluated using the anti-attack rate (AAR). The total number of training iterations is 500, and the gradient attack is executed in every iter; once the fitted gradient value is less than 1, the gradient reconstruction is successful, and the sample of input can be recovered. For each attack, including completely failed gradient attacks, the predefined attack limit is set to 300. We recorded the changes in the gradient reconstruction procedure.

The heat maps in FIGURE 7 record the variation in the reconstructed gradients with the number of attacks under the gradient attack approach for various perturbation strategies in training.



(a) no noise

(b) noise scale σ=1

(c) noise scale σ=[1, 2]

(d) bilateral alternative noise, scale σ=[1, 2]

FIGURE 7. Gradients attack in various noise strategy.

TABLE 3. Resilience against gradients attack (training iterations=500).

method	noise scale	success	failure	Anti-attack rate
no noise	-	433	67	13.4%
fixed σ	1	197	303	60.6%
dynamic σ	[1, 2]	175	325	65%
bilateral dynamic σ	[1, 2]	0	500	100%

Table 3 provides a quantitative analysis of the resistance to invasion. Combined with the graphs, when there is no privacy scenario, most of the attacks in FIGURE 7(a) can restore the sample, and the number of successful gradient reconstructions is always all less than 30 rounds, accounting for less than 1/10 of the predefined upper limit. When (ii) shown in FIGURE 7(b) and (iii) shown in FIGURE 7(c) are used to protect the gradients during training, the gradient values of the reconstruction attack become significantly more volatile and the success of the gradient assault becomes less easy. The overall noise scale with a dynamic value domain is larger. Therefore, it requires more rounds than the fixed noise scale because gradient reconstructions are expected to be successful. The overall constructed gradient fluctuation was also larger.

Table 3 shows that the AAR with fixed σ is 60.6%, and that with dynamic σ is 65%, which are 352.2% and 385.1% higher than the anti-attack rate without noise (13.4%), respectively. When we use (iv) shown in FIGURE 7(d), the gradient reconstruction process oscillates violently and is absolutely unable to meet the demand of recovering the sample, and the gradient intrusion fails completely.

Larger range-valued noise provides a more robust defensive barrier than conventional fixed noise scales. However, in a summary of the strategies demonstrated above, the gradient privacy-preserving training methods of gradient attack defense using our bilateral alternative dynamic range-valued noise work best, mainly attributed to the combined effect of greater sensitivity and larger noise scale. Effectively makes it difficult for a successful gradient attack to occur, and defending a privacy breach.

VI. CONCLUSION

Our DPBA-WGAN adds a specially designed VVG noise to the critic vector valued cost function. The methodology can indirectly transform gradients to a non-sensitive numerical type, ensuring that the generative model can produce expected amounts of synthetic dataset, which matches the statistical properties of the source data and does not compromise privacy.

The DP training strategy in our algorithm aims at an overall network architecture with noise-free decay, to participate in parametric optimization and interaction. We do not change the internal structure of the discriminator and generator, which makes the training process considerably easier and faster. Considering the bounded cost and gradients by the Lipschitz condition, we accurately estimate the sensitivity in differential privacy. The restriction prevents the gradient from exploding or diminishing. Using the Wasserstein distance to train the WGAN network, which can eliminate the pitfall of poor data generation owing to the instability of the network. The evaluation of datasets with different numbers of channels shows that the datasets generated with DPBA have excellent generation quality and are well-suited for continuous commitment to the use of downstream tasks. In summary, the novel strategy provides a new reference for the privacy-preserving training of deep learning models and data generations.

APPENDIX A
RENYI DIFFERENTIAL PRIVACY PROOF AND BUDGET

Based on the well-defined sensitivity and the known sampling rate r ($r = B/A$), where B denotes the batch size, and A denotes all sample sizes, a Gaussian mechanism satisfying (α, ε)-differential privacy can be designed.

Theory 1. Given the sampling rate r and sensitivity ΔS, for any positive number $\alpha \geq 1$, add the probability density function as $\mathcal{N}\left(\mu, \sigma^2\right)$ random noise to the vector loss function during each round of WGAN training, then the mechanism satisfies (α, ε)-RDP and the level of privacy protection is $\varepsilon \leq r\alpha^2 \Delta S^2 / \left[2\sigma^2 (\alpha - 1)\right]$.

It is then shown that after T rounds of iterative training, the parameter values and loss still guarantee differential privacy, and a more conventional variant of $\left(\varepsilon', \delta\right)$ can be obtained by applying the idea from Corollary 1. That is, given a training number T, a sampling rate r and sensitivity ΔS of the dataset, for any positive number $\alpha \geq 1$, $0 < \delta < 1$, such that $\log\left(1/\delta\right) \geq 2T$, a random noise with pdf $\mathcal{N}\left(\mu, \sigma^2\right)$ is added to the target vector function or parameters during WGAN training, making the mechanism a T $\left(\varepsilon', \delta\right)$-DP component theory mechanism, where $\varepsilon = \left(2r\alpha^2\Delta S^2\right) / \left[(\alpha - 1)\sigma^2\right]$, so that we have:

$$\varepsilon' = \left(r\alpha^2 \Delta S^2\right) \Big/ \left[2(\alpha - 1)\sigma^2\right] \sqrt{2T \log\left(1/\delta\right)} \qquad (14)$$

From Equation (14), it can be shown that the discriminator satisfies $\left(\varepsilon', \delta\right)$-differential privacy.

For the sake of the following argumentative illustration, the coefficient $\alpha^2/(\alpha-1) \geq 4$, there exists a number greater than or equal to $4\sqrt{2}/\sigma$, denoted as α', such that $\varepsilon' = \left(\alpha'/\sigma\right) 2r\sqrt{T \log\left(1/\delta\right)}$.

proof. The target distribution to be perturbed is denoted by $\mathcal{L}_D$, and X and X' are denoted as two adjacent datasets that differ by at most one element from each other. Therefore, after adding noise to one part of the cost function for perturbation, the loss functions of the two adjacent datasets can be expressed as follows:

$$\begin{aligned} \mathcal{L}_D\left(X\right) &= \mathcal{L}_D\left(\theta, X\right) + \mathcal{N}\left(\mu, \sigma^2\right) \\ \mathcal{L}'_D\left(X'\right) &= \mathcal{L}_D\left(\theta, X'\right) + \mathcal{N}\left(\mu, \sigma^2\right) \end{aligned} \tag{15}$$

where $\mathcal{L}_D\left(\theta, X\right) = \frac{1}{B}\left(\sum_{i=1, i\neq j}^{B} \mathcal{M}\left(x_i\right) + \mathcal{M}\left(x_j\right)\right)$, and $\mathcal{L}'_D\left(\theta, X'\right) = \frac{1}{B}\left(\sum_{i=1, i\neq j}^{B} \mathcal{M}\left(x_i\right) + \mathcal{M}\left(x'_j\right)\right)$, θ denote various types of parameters in the training process of the network.

In Equation (15), we know that $\mathcal{L}_D\left(X\right)$ is the sum of two independent variables: $\mathcal{L}_D\left(\theta, X\right)$ and $\mathcal{N}\left(\mu, \sigma^2\right)$. According to the Gaussian approximation principle, the final function is obtained by adding two independent variables that conform to the Gaussian distribution; thus, $\mathcal{L}_D\left(X\right)$ also conforms to the Gaussian distribution. Similarly, $\mathcal{L}'_D\left(X'\right)$ obeys a Gaussian distribution. Without loss of generality, assume that the two distributions, $\mathcal{L}_D\left(X\right)$ and $\mathcal{L}'_D\left(X'\right)$, differ only in the first characteristic, we define $u_0\mathcal{N}\left(\mu, \sigma^2\right)$, $u_1\mathcal{N}\left(\mu + \Delta S, \sigma^2\right)$, with:

Case 1. supposing that

$$\mathcal{L}_D\left(X\right) \sim \left(1-r\right) u_0 + r u_1 \mathcal{L}'_D\left(X'\right) \sim u_0\mathcal{N}\left(\mu, \sigma^2\right)$$

For simplicity and clarity, the expectation μ is set on the y-axis and according to the definition of Renyi entropy, it is derived that,

$$\begin{aligned} &D_\alpha\left[\mathcal{L}_D\left(X\right) || \mathcal{L}'_D\left(X'\right)\right] \\ &= \frac{1}{\alpha-1} \log \int \left[\left(1-r\right) u_0 + r u_1\right]^{\alpha} u_0^{1-\alpha} dx \\ &= \frac{1}{\alpha-1} \log \int u_0^{1-\alpha} \sum_{k=0}^{\alpha} \binom{k}{\alpha} \left(r u_1\right)^k \left[\left(1-r\right) u_1\right]^{\alpha-k} dx \\ &= \frac{1}{\alpha-1} \log \int u_0 \sum_{k=0}^{\alpha} \binom{k}{\alpha} r^k \left(1-r\right)^{\alpha-k} e^{-\frac{k\left(\Delta S^2 - 2\Delta S x\right)}{2\sigma^2}} dx \\ &= \frac{1}{\alpha-1} \log \int u_0 \sum_{k=0}^{\alpha} r^k \left(1-r\right)^{\alpha-k} e^{\frac{k^2\Delta S^2 - k\Delta S^2}{2\sigma^2}} \frac{1}{\sigma\sqrt{2\pi}} \\ &\quad \times e^{-\frac{\left(x - k\Delta S\right)^2}{2\sigma^2}} dx \\ &= \frac{1}{\alpha-1} \log \sum_{k=0}^{\alpha} r^k \left(1-r\right)^{\alpha-k} e^{\frac{k\left(k-1\right)\Delta S^2}{2\sigma^2}} \\ &\leq \frac{1}{\alpha-1} \log \left(1 - r + re\frac{\alpha\Delta S^2}{2\sigma^2}\right) \\ &= \frac{\alpha}{\alpha-1} \log \left(1 - r + re^{\frac{\alpha\Delta S^2}{2\sigma^2}}\right) \\ &= r\frac{\alpha}{\alpha-1}\left(e^{\frac{\alpha\Delta S^2}{2\sigma^2}} - 1\right) + o\left(\left(\frac{\alpha\Delta S^2}{2\sigma^2}\right)^2\right) \end{aligned} \tag{16}$$

Case 2. supposing that

$$\mathcal{L}_D\left(X\right) \sim u_0\mathcal{N}\left(\mu, \sigma^2\right) \mathcal{L}'_D\left(X'\right) \sim \left(1-r\right) u_0 + r u_1$$

Since we have:

$$\begin{aligned} &\frac{1}{\alpha-1} \log \int u_1 \left[1 - r + r\frac{u_1}{u_0}\right]^{\alpha} \\ &= \frac{1}{\alpha-1} \log \int \sum_{k=0}^{\alpha} \binom{k}{\alpha} r^k \left(1-r\right)^{\alpha-k} e^{\frac{k\left(k+1\right)\Delta S^2}{2\sigma^2}} \frac{1}{\sigma\sqrt{2\pi}} \\ &\quad \times e^{-\frac{\left[x - \left(k+1\right)\Delta S\right]^2}{2\sigma^2}} \\ &= \frac{1}{\alpha-1} \log \sum_{k=0}^{\alpha} \binom{k}{\alpha} r^k \left(1-r\right)^{\alpha-k} e^{\frac{k\left(k+1\right)\Delta S^2}{2\sigma^2}} \\ &\leq \frac{1}{\alpha-1} \log \left[1 - r + re^{\frac{\left(\alpha+1\right)\Delta S^2}{2\sigma^2}}\right]^{\alpha} \\ &= r\frac{\alpha\left(\alpha+1\right)\Delta S^2}{2\sigma^2\left(\alpha-1\right)} + o\left(\left(\frac{\left(\alpha+1\right)\Delta S^2}{2\sigma^2}\right)^2\right) \end{aligned} \tag{17}$$

following the definition of Renyi entropy, it is concluded that,

$$\begin{aligned} &D_\alpha\left[\mathcal{L}_D\left(X\right) || \mathcal{L}'_D\left(X'\right)\right] \\ &= \frac{1}{\alpha-1} \log \int u_0^{\alpha} \left[\left(1-r\right) u_0 + r u_1\right]^{1-\alpha} dx \\ &= \frac{1}{\alpha-1} \log \int \left[\left(1-r\right) u_0 + r u_1\right] \left[\frac{u_0}{\left(1-r\right) u_0 + r u_1}\right]^{\alpha} dx \\ &= \frac{1}{\alpha-1} \log \int \left[\left(1-r\right) u_0 \left(1 - r + r\frac{u_1}{u_0}\right)^{-\alpha}\right. \\ &\quad \left. + r u_1 \left(1 - r + r\frac{u_1}{u_0}\right)^{-\alpha}\right] dx \\ &\leq \left(1-r\right) r\frac{\alpha^2\Delta S^2}{2\sigma^2\left(\alpha-1\right)} + r^2\frac{\alpha\Delta S^2}{2\sigma^2} + o\left(\left(\frac{\alpha\Delta S^2}{2\sigma^2}\right)^2\right) \end{aligned} \tag{18}$$

where $\frac{1}{\alpha-1} \log \int u_0 \left(1 - r + r\frac{u_1}{u_0}\right)^{-\alpha} dx \leq r\frac{\alpha^2\Delta S^2}{2\sigma^2(\alpha-1)} + o\left(\left(\frac{\alpha\Delta S^2}{2\sigma^2}\right)^2\right)$ is determined by (16), $\frac{1}{\alpha-1} \log \int u_1 \left(1 - r + r\frac{u_1}{u_0}\right)^{-\alpha} dx \leq r\frac{\alpha\Delta S^2}{2\sigma^2} + o\left(\left(\frac{(\alpha-1)\Delta S^2}{2\sigma^2}\right)^2\right)$ is arrived at based on (17).

It is obvious that $\frac{\alpha\Delta S^2}{2\sigma^2} < \frac{\alpha^2\Delta S^2}{2\sigma^2(\alpha-1)}$ is from (18), and so $\left(1-r\right) r\frac{\alpha^2\Delta S^2}{2\alpha^2(\alpha-1)} + r^2\frac{\alpha\Delta S^2}{2\sigma^2} < r\frac{\alpha^2\Delta S^2}{2\sigma^2(\alpha-1)}$. The ultimate

TABLE 4. Accuracy in multiple classifiers on MNIST ($\delta = 10^{-5}$).

		Real	DP-GAN	DP-MERF	GS-WGAN	DP-Sinkhorn	Ours
DP-ε		∞	10	1	10	10	1
FID↓		1.0238	354.6802	351.2738	61.34	23.66297	4.4665
cnn	acc↑	0.9885	0.6931	0.7574	0.8	0.883	0.9704
mlp		0.9758	0.574	0.7651	0.79	0.7628	0.9338
logistic_reg		0.9256	0.4839	0.7959	0.79	0.7247	0.8758
random_forest		0.9699	0.4473	0.2887	0.52	0.1751	0.7456
gaussian_nb		0.5558	0.1064	0.7078	0.64	0.5069	0.5792
bernoulli_nb		0.8427	0.5836	0.7023	0.77	0.732	0.8392
linear_svc		0.9215	0.5727	0.7676	0.76	0.7135	0.8718
decision_tree		0.8808	0.2806	0.1871	0.35	0.2519	0.462
lda		0.8798	0.4852	0.8049	0.78	0.7177	0.8662
adaboost		0.7296	0.1379	0.1889	0.21	0.1164	0.2278
bagging		0.9265	0.2987	0.2678	0.45	0.1684	0.4957
gbm		0.9085	0.2302	0.4866	0.39	0.1864	0.6724
xgboost		0.9736	0.3798	0.1032	0.5	0.1707	0.2142
average		0.8830	0.4056	0.5249	0.5962	0.4670	0.6734

TABLE 5. Accuracy in multiple classifiers on Fashion-MNIST ($\delta = 10^{-5}$).

		Real	DP-GAN	DP-MERF	GS-WGAN	DP-Sinkhorn	Ours
DP-ε		∞	10	1	10	10	1
FID↓		1.4906	370.7651	309.5515	131.34	29.5840	28.1722
cnn	acc↑	0.9099	0.644	0.6468	0.64	0.7657	0.8091
mlp		0.8779	0.5784	0.6608	0.65	0.7529	0.7726
logistic_reg		0.8441	0.5903	0.6675	0.68	0.7311	0.7477
random_forest		0.8778	0.5953	0.4807	0.54	0.4375	0.5129
gaussian_nb		0.5856	0.1671	0.6275	0.48	0.5298	0.6092
bernoulli_nb		0.648	0.5508	0.5829	0.55	0.5566	0.6303
linear_svc		0.8397	0.5878	0.7126	0.65	0.7126	0.7258
decision_tree		0.7896	0.3232	0.1867	0.4	0.3224	0.4241
lda		0.7996	0.6279	0.6753	0.67	0.7575	0.7605
adaboost		0.5618	0.1539	0.28	0.25	0.3258	0.4128
bagging		0.8452	0.3493	0.2856	0.47	0.4056	0.3822
gbm		0.8331	0.3557	0.4521	0.38	0.4294	0.564
xgboost		0.8825	0.5333	0.2777	0.47	0.5036	0.3216
average		0.7919	0.4659	0.5028	0.5254	0.5562	0.5902

calculation is $\varepsilon = r\frac{\alpha^2 \Delta S^2}{2\sigma^2(\alpha-1)}$, which satisfies (α, ε)-RDP. Similarly, it can be proven that the parameters also satisfy (α, ε)-RDP, and the whole WGAN satisfies RDP by differential privacy transferability. Hence, the fictitious data released by the generator can achieve the purpose of protect the privacy of the source data.

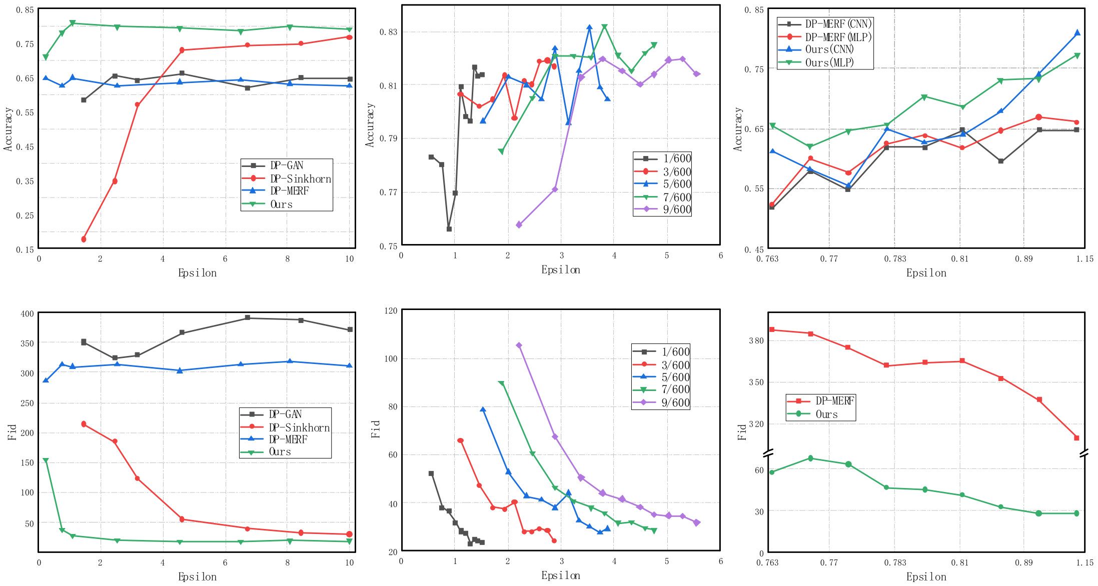
APPENDIX B MORE DETAILS OF DATASETS

A. MNIST

This dataset contains statistics consisting of 250 handwritten digits from different people with four files: training images, training labels, test images, and test labels. The training set contained 60,000 images and labels, whereas the test set contained 10,000 images and labels. The anterior 5,000 in the test set were from the training set of the original NIST project and the posterior 5,000 were from the test set of the original NIST project. The size of each image was 28 × 28 and each label was a one-dimensional array of length 10. In the deep-learning domain, handwritten digit recognition is an essential example of learning.

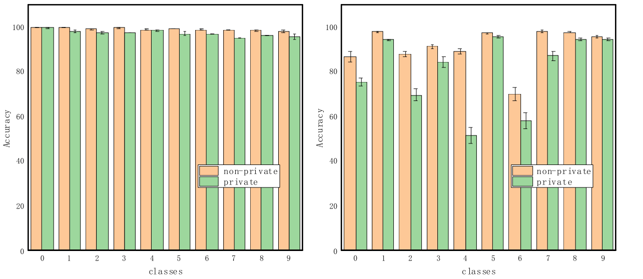
B. FASHION-MNIST

It is slightly more complex than MNIST and consists of 10 categories of apparel: t-shirt, trouser, pullover, dress, coat, sandal, shirt, sneaker, bag, and ankle boot. It contains 70,000



(a) Iteration influences (b) Subsampling rate influences (c) Noise scale influences

FIGURE 8. Analysis of hyperparameters on Fashion-Mnist($\delta = 10^{-5}$).



(a) Accuracy by category on MNIST (b) Accuracy by category on Fashion-MNIST

FIGURE 9. Classification accuracy of downstream task between non-private and private data($\varepsilon = 1, \delta = 10^{-5}$).

images, of which 60,000 are training images and 10,000 are test images. The image size is 28 × 28, which are single channels.

C. SVHN

Since the algorithm that performs better in the Mnist and Fashion-Mnist datasets with simpler features may not be applicable to other sophisticated tasks, we also used the SVHN dataset. Compared to the two datasets mentioned above, the dataset used in our experiment is a single-numbered SVHN with a pre-process, which is a 3-channel RGB image. The SVHN has an image size of 32 × 32, which is marginally larger than that of the previous two datasets. Compared with handwritten characters, SVHN originates from real scenes in life, which are not only noisy but also have diverse proportions and features of objects, which poses great challenges for recognition by machines.

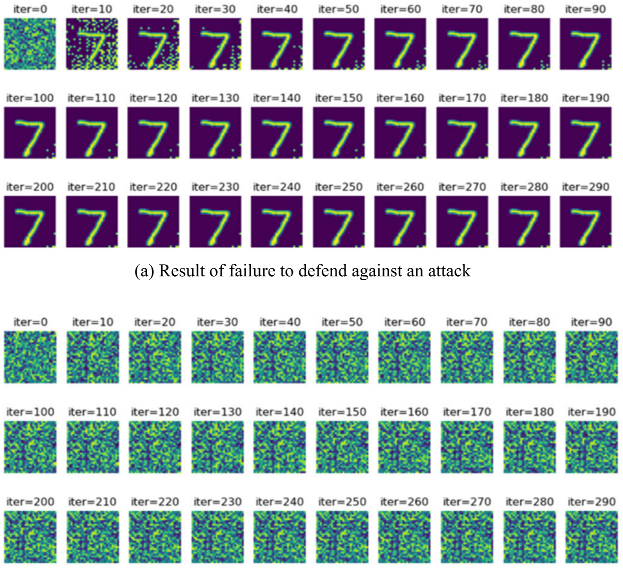
D. CelebA

There are three types of files in this dataset: pure "wild" files, which are images crawled from the web without cropping, images after cropping out the face part from "wild" files; and cropped face images in jpg format. In our experiment, we used a lightweight jpg format file. There are more than 200,000 images, including 40 attributes, such as gender, beard, hair style, hair color, and skin tone. We only need to consider whether the algorithm is also feasible for RGB images and the dichotomous situation in the comparison experiment. Therefore, so in order to speed up the experiment progress, 25,156 male images and 28,234 female images were filtered through the dataset labels of the attribute file species for generating network training, and 10% of the dataset was left to test the effectiveness of the yielding images.

APPENDIX C
ADDITIONAL EXPERIMENTAL RESULTS

To supplement the experimental findings in TABLE 1, TABLE 4, TABLE 5, the results of Mnist and Fashion-Mnist on multiple downstream classifiers are presented.

All accuracy values in the tables were obtained using the early stop training method and recording the average of five testing outcomes. FIGURE 8 shows the hyperparameter analysis on Fashion-Mnist to evaluate the privacy and usability of this dataset when trained on the differential privacy WGAN network. The same three hyperparameters are used as variables in FIGURE 4. The analysis of various datasets demonstrates the generalization ability of the proposed scheme. FIGURE 9 shows the accuracy corresponding to each attribute after downstream classification by adopting our generated method, and a comparison with the approach without a privacy mechanism shows that the generative model is balanced for all types of objects. The accuracy and error of the histogram were averaged over five trials using early stop.



(a) Result of failure to defend against an attack

(b) Result of a well-conceived privacy policy

FIGURE 10. Visualization of results under gradient invasion.

FIGURE 10 is used to supplement the results of the ability to defend against different noise approaches when subjected to a gradient attack, and is the result of the image visualization reconstructed from the intruder gradient attack. We select the process of recovering the image when subjected to an attack in two certain training rounds, and in the results shown in the figure, if the attack fails, the recovered image is all of the noise.

REFERENCES

[1] Y. He, G. Meng, K. Chen, X. Hu, and J. He, "Towards security threats of deep learning systems: A survey," *IEEE Trans. Softw. Eng.*, vol. 48, no. 5, pp. 1743–1770, May 2022.

[2] M. E. Gursoy, A. Inan, M. E. Nergiz, and Y. Saygin, "Differentially private nearest neighbor classification," *Data Mining Knowl. Discovery*, vol. 31, no. 5, pp. 1544–1575, Sep. 2017.

[3] H. Liu, Z. Q. Wu, C. G. Peng, F. Tian, and L. F. Lu, "Adaptive Gaussian mechanism based on expected data utility under conditional filtering noise," *KSII Trans. Internet Inf. Syst.*, vol. 12, no. 7, pp. 3497–3515, Jul. 2018.

[4] Y. Li, X. Ren, F. Zhao, and S. Yang, "P-power exponential mechanisms for differentially private machine learning," *IEEE Access*, vol. 9, pp. 155018–155034, 2021.

[5] M. Park, J. Foulds, K. Chaudhuri, and M. Welling, "Variational Bayes in private settings (VIPS)," *J. Artif. Intell. Res.*, vol. 68, pp. 109–157, May 2020.

[6] Z. Sun, Y. Wang, M. Shu, R. Liu, and H. Zhao, "Differential privacy for data and model publishing of medical data," *IEEE Access*, vol. 7, pp. 152103–152114, 2019.

[7] A. Imakura and T. Sakurai, "Data collaboration analysis framework using centralization of individual intermediate representations for distributed data sets," *ASCE-ASME J. Risk Uncertainty Eng. Syst., A, Civil Eng.*, vol. 6, no. 2, Jun. 2020.

[8] K. Bonawitz, P. Kairouz, B. McMahan, and D. Ramage, "Federated learning and privacy," *Commun. ACM*, vol. 65, no. 4, pp. 90–97, Apr. 2022.

[9] P. Li, T. Li, H. Ye, J. Li, X. Chen, and Y. Xiang, "Privacy-preserving machine learning with multiple data providers," *Future Gener. Comput. Syst.*, vol. 87, pp. 341–350, Oct. 2018.

[10] X. Huang, Y. Ding, Z. L. Jiang, S. Qi, X. Wang, and Q. Liao, "DP-FL: A novel differentially private federated learning framework for the unbalanced data," *World Wide Web*, vol. 23, no. 4, pp. 2529–2545, Jul. 2020.

[11] L. Lyu, Y. Li, K. Nandakumar, J. Yu, and X. Ma, "How to democratise and protect AI: Fair and differentially private decentralised deep learning," *IEEE Trans. Depend. Secure Comput.*, vol. 19, no. 2, pp. 1003–1017, Mar./Apr. 2020.

[12] *Apple Differential Privacy Technical Overview*. Accessed: Jun. 13, 2016. [Online]. Available: https://www.apple.com/privacy/docs/Differential_Privacy_Overview.pdf

[13] M. Hay, M. Gaboardi, and S. Vadhan, "A programming framework for OpenDP," in *Proc. 6th Workshop Theory Pract. Differential Privacy*, 2020, pp. 1–63.

[14] A. Bavadekar et al., "Google COVID-19 vaccination search insights: Anonymization process description," Jul. 2021.

[15] Y. Wang, M. Gu, J. Ma, and Q. Jin, "DNN-DP: Differential privacy enabled deep neural network learning framework for sensitive crowdsourcing data," *IEEE Trans. Computat. Social Syst.*, vol. 7, no. 1, pp. 215–224, Feb. 2020.

[16] F. Farokhi, "Privacy-preserving public release of datasets for support vector machine classification," *IEEE Trans. Big Data*, vol. 7, no. 5, pp. 893–899, Nov. 2021.

[17] S. Ho, Y. Qu, B. Gu, L. Gao, J. Li, and Y. Xiang, "DP-GAN: Differentially private consecutive data publishing using generative adversarial nets," *J. Netw. Comput. Appl.*, vol. 185, Jul. 2021, Art. no. 103066.

[18] S. Zhang, W. Ni, and N. Fu, "Differentially private graph publishing with degree distribution preservation," *Comput. Secur.*, vol. 106, Jul. 2021, Art. no. 102285.

[19] B. Xin, Y. Geng, T. Hu, S. Chen, W. Yang, S. Wang, and L. Huang, "Federated synthetic data generation with differential privacy," *Neurocomputing*, vol. 468, pp. 1–10, Jan. 2022.

[20] B.-W. Tseng and P.-Y. Wu, "Compressive privacy generative adversarial network," *IEEE Trans. Inf. Forens. Security*, vol. 15, pp. 2499–2513, 2020.

[21] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2016, pp. 308–318, doi: 10.1145/2976749.2978318.

[22] J. Yang, L. Xiang, R. Chen, W. Li, and B. Li, "Differential privacy for tensor-valued queries," *IEEE Trans. Inf. Forensics Security*, vol. 17, pp. 152–164, 2022.

[23] W. Wei and L. Liu, "Gradient leakage attack resilient deep learning," *IEEE Trans. Inf. Forensics Security*, vol. 17, pp. 303–316, 2022.

[24] L. Xie, K. Lin, S. Wang, F. Wang, and J. Zhou, "Differentially private generative adversarial network," 2018, *arXiv:1802.06739*.

[25] L. Frigerio, A. S. D. Oliveira, L. Gomez, and P. Duverger, "Differentially private generative adversarial networks for time series, continuous, and discrete open data," in *Proc. IFIP Int. Conf. ICT Syst. Secur. Privacy Protection*, vol. 562. Cham, Switzerland: Springer, 2019, pp. 151–164.

[26] R. Torkzadehmahani, P. Kairouz, and B. Paten, "DP-CGAN: Differentially private synthetic data and label generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 98–104.

[27] D. F. Chen, T. Orekondy, and M. Fritz, "GS-WGAN: A gradient-sanitized approach for learning differentially private generators," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 12673–12684.

[28] J. Jordon, J. Yoon, and M. V. D. Schaar, "PATE-GAN: Generating synthetic data with differential privacy guarantees," presented at the Int. Conf. Learn. Represent., 2018. [Online]. Available: https://openreview.net/forum?id=S1zk9iRqF7

[29] F. Harder, K. Adamczewski, and M. Park, "Differentially private mean embeddings with random features (DP-MERF) for simple & practical synthetic data generation," in *Proc. 24th Int. Conf. Artif. Intell. Statist.*, vol. 130, 2021, pp. 1819–1827.

[30] T. S. Cao, A. Bie, A. Vahdat, S. Fidler, and K. Kreis, "Don't generate me: Training differentially private generative models with sinkhorn divergence," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 12480–12492.

[31] M. Arjovsky and L. Bottou, "Towards principled methods for training generative adversarial networks," *Stat*, vol. 1050, no. 17, pp. 1–17, 2017.

[32] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 214–223.

[33] C. Han and R. Xue, "Differentially private GANs by adding noise to discriminator's loss," *Comput. Secur.*, vol. 107, Aug. 2021, Art. no. 102322.

[34] C. Dwork, "Differential privacy," in *Automata, Languages and Programming*, vol. 4052, M. Bugliesi, B. Preneel, V. Sassone, I. Wegener, Eds. Berlin, Germany: Springer, 2006, pp. 1–12.

[35] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor, "Our data, ourselves: Privacy via distributed noise generation," in *Proc. Annu. Int. Conf. Theory Appl. Cryptograph. Techn.*, vol. 4004. Cham, Switzerland: Springer, 2006, p. 486.

[36] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Found. Trends Theor. Comput. Sci.*, vol. 9, nos. 3–4, pp. 211–407, 2014.

[37] I. Mironov, "Rényi differential privacy," in *Proc. IEEE 30th Comput. Secur. Found. Symp. (CSF)*, Aug. 2017, pp. 263–275.

[38] I. Goodfellow, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 63, no. 11, 2014, pp. 139–144.

[39] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of Wasserstein GANs," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 5769–5779.

[40] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 2180–2188.

[41] E. Schonfeld, B. Schiele, and A. Khoreva, "A U-Net based discriminator for generative adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8207–8216.

[42] Y. Feigin, H. Spitzer, and R. Giryes, "GMM-based generative adversarial encoder learning," 2020, *arXiv:2012.04525*.

[43] S. Suh, J. Kim, P. Lukowicz, and Y. O. Lee, "Two-stage generative adversarial networks for binarization of color document images," *Pattern Recognit.*, vol. 130, Oct. 2022, Art. no. 108810.

[44] J. Dong, A. Roth, and W. J. Su, "Gaussian differential privacy," *J. Roy. Stat. Soc., Ser. B, Stat. Methodol.*, vol. 84, no. 1, pp. 3–37, Feb. 2022.

[45] K. Wei, "Performance analysis and optimization in privacy-preserving federated learning," Feb. 2020, *arXiv:2003.00229*.

[46] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[47] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," 2017, *arXiv:1708.07747*.

[48] Y. Netzer, T. Wang, A. Coates, A. Bissacco, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *Proc. NIPS Workshop Deep Learn. Unsupervised Feature Learn.*, 2011, pp. 1–9.

[49] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3730–3738.

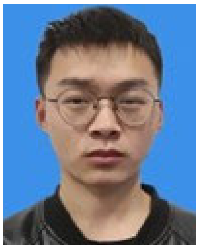
[50] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local Nash equilibrium," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 6629–6640.

[51] H. Liang, S. Zhang, J. Sun, X. He, W. Huang, K. Zhuang, and Z. Li, "DARTS+: Improved differentiable architecture search with early stopping," 2019, *arXiv:1909.06035*.

[52] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–26.

[53] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *Proc. Int. Conf. Learn. Represent.*, 2016, pp. 1–16.

DANHUA WU was born in Zhejiang, China, in 1996. She received the B.S. degree in computer science and technology from the Tianjin University of Commerce, in 2019. She is currently pursuing the M.S. degree with the Department of Information Science and Technology, Guilin University of Technology, China. Her research interests include artificial intelligence and privacy protection.



WENYONG ZHANG was born in Jiangxi, China, in 1998. He received the B.S. degree in computer science and technology from Beijing Union University, in 2020. He is currently pursuing the M.S. degree with the Department of Information Science and Technology, Guilin University of Technology, China. His research interest includes machine learning with privacy protection.

PANFENG ZHANG was born in Hubei, China, in 1978. He received the Ph.D. degree from the Huazhong University of Science and Technology, China. Since 2017, he has been a Lecturer at the Department of Information Science and Technology, Guilin University of Technology. He is currently the Host of Research on Hierarchical Diversity Anonymous Method for Data Publishing Privacy Protection sponsored by the National Natural Science Foundation of China. His research interests include information storage, information security, and artificial intelligence technology.