Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 6680701, 20 pages
https://doi.org/10.1155/2021/6680701

Research Article

Face Image Publication Based on Differential Privacy

Chao Liu,[1,2] **Jing Yang**,[1] **Weinan Zhao**,[2] **Yining Zhang**,[3] **Jingyou Li**,[1,2] **and Chunmiao Mu**[2]

[1]*Harbin Engineering University, Harbin 150001, China*
[2]*Qiqihar University, Qiqihar 161000, China*
[3]*DaQing Vocational College, DaQing 163000, China*

Correspondence should be addressed to Jing Yang; 00819@qqhru.edu.cn

Received 9 October 2020; Revised 28 November 2020; Accepted 10 December 2020; Published 7 January 2021

Academic Editor: Chi-Hua Chen

As an information carrier, face images contain abundant sensitive information. Due to its natural weak privacy, direct publishing may divulge privacy. Anonymization Technology and Data Encryption Technology are limited by the background knowledge and attack means of attackers, which cannot completely content the needs of face image privacy protection. Therefore, this paper proposes a face image publishing SWP (sliding window publication) algorithm, which satisfies the differential privacy. Firstly, the SWP translates the image gray matrix into a one-dimensional ordered data stream by using image segmentation technology. The purpose of this step is to transform the image privacy protection problem into the data stream privacy protection problem. Then, the sliding window model is used to model the data flow. By comparing the similarity of data in adjacent sliding windows, the privacy budget is dynamically allocated, and Laplace noise is added. In SWP, the data in the sliding window comes from the image. To present the image features contained in the data more comprehensively and use the privacy budget more reasonably, this paper proposes a fusion similarity measurement EM (exact mechanism) mechanism and a dynamic privacy budget allocation DA (dynamic allocation) mechanism. Also, for further improving the usability of human face images and reducing the impact of noise, a sort-SWP algorithm based on the SWP method is proposed in the paper. Through the analysis, it can be seen that ordered input can further improve the usability of the SWP algorithm, but direct sorting of data will destroy the ε-differential privacy. Therefore, this paper proposes a sorting method-*SAS* method, which satisfies the ε-differential privacy; SAS obtain an initial sort by using an exponential mechanism firstly. And then an approximate correct sort is obtained by using the Annealing algorithm to optimize the initial sort. Compared with LAP algorithm and SWP algorithm, the average accuracy rate of sort-SWP algorithm in ORL, Yale is increased by 56.63% and 21.55%, the recall rate is increased by 6.85% and 3.32%, and F1-sroce is improved by 55.62% and 16.55%.

1. Introduction

With the rapid development of information technology and multimedia technology, it is easier to obtain and share face digital images. Users can publish photos of their mobile phones or digital cameras to social networking platforms (such as Twitter, LinkedIn, WeChat) or other channels. Relevant statistics show that the number of face photos shared by users on major social networking platforms worldwide exceeds 3.2 billion every day. Also, there are numerous face image data derived from video. However, these digital images usually contain a wealth of personally sensitive information. If this information was collected and analyzed by a third party with ulterior motives, it may cause personal privacy disclosure and other unexpected losses.

Privacy is an emotional word, which implies different meanings to different people. According to the definition of International Organization Standardization (ISO), privacy refers to the characteristics that can distinguish individuals or groups from other individuals and groups. Different countries have different legal definitions of privacy, and different objects (individuals, enterprises, governments, etc.) define

the scope of privacy differently. For digital images, sensitive information can be a specific person or object in an image, a face, or fingerprint; the embedded information (photo location information and creation time, etc.); or an area that the image owner is concerned about. How to publish and analyze without disclosing sensitive information is the main purpose of privacy protection.

The early research uses Anonymization Technology or Data Encryption Technology to solve the privacy protection of face image. Anonymization Technology refers to cover up real data with methods of hidden or fuzzy. It generally adopts anonymous operations such as suppression [1], generalization [2], analysis [3], slicing [4], and separation [5]. Especially, k-anonymity [6] is a classic representative algorithm. K-anonymity proposed that the sensitive information covered by the data should at least be indistinguishable from other $k-1$ data. Because of its shortcomings and shortcomings, K-anonymity extends the l-diversity method which ensures that each equivalent class contains at least l different sensitive attribute values [7], the t-closeness method to improve the global distribution of sensitive attributes [8], the m-variance method for dynamic relational data [9], and the HD composition method [10]. References [11, 12] use an anonymization mechanism to propose the k-same method. This method anonymizes the published digital image so that the probability of the attacker to identify the user identity through the published digital image again is less than $1/K$. However, the main drawback of the traditional anonymization mechanism is too many assumptions about the attacker's background knowledge and attack model, but those assumptions are not completely successful in reality.

Data Encryption Technology is another important research direction of image information security, and its representative methods include secure multiparty computing [13], homomorphism encryption [14], and classification algorithm [15]. References [16–19] prevent the invasion of the third party by controlling the user communication protocol. References [20–23] propose using pixel replacement or pixel value substitution to encrypt image content. References [24–28] encrypt the image by changing the transform coefficient of the image in the frequency domain. Similar to the problems of anonymization technology, data encryption technology will also make corresponding assumptions for attacks and then design the corresponding encryption algorithm based on these assumptions. But this kind of encryption method will fall into the cycle of "new encryption methods are constantly proposed but constantly broken." Also, the encrypted image is not open.

Dwork first proposed differential privacy [29] in 2006, which disturbs sensitive data by adding noise to the output. Differential privacy can hide the influence of a single record. That means whether the record is in or not in the dataset, the output probability of the same result will not change significantly. The attacker's ability to further reasoning is limited. Therefore, differential privacy is better than other privacy protection technologies by not making any assumptions about the background knowledge of any potential attacker. Besides, differential privacy is further studied in a series of papers of Dwork [30–34], and its implementation mechanism is proposed in [35, 36]. McSherry pointed out that some differential privacy algorithms for complex privacy problems satisfy two combinatorial properties: sequence composability and juxtaposition and combination [37]. In recent years, differential privacy is mainly used in data publishing, including histogram publishing [38–43], graph data publishing [44–48], data mining [49–51], data stream publishing [52], and spatial data publishing [53]. Due to the complexity of image data, researchers are still in the exploratory stage to use differential privacy technology to protect sensitive information in images.

The real field matrix is a common representation of the image. Any pixel in the image can be mapped to a numerical relative position in a 2D matrix. It is the most direct method to add Laplace noise to all the values in the matrix. Although this method can satisfy ε-differential privacy, it will cause excessive distortion and low usability of the disturbing image. Fourier transform and Wavelet transform are commonly compression techniques in image processing. In reference [54], an image compression method based on a discrete Fourier transform is proposed. This method adds the corresponding Laplace noise to the compressed image. Although the noise error is reduced, the reconstruction error is introduced in the image compression process. To reduce the impact of noise on the original image and improve the usability of published images, this paper proposes a differential privacy protection method for image publishing. It is inspired by the noncorrelation of the values in the image matrix; this paper tries to use image segmentation technology to transform the image gray matrix into 1D ordered data stream and then use the sliding window model to model the data flow. By comparing the similarity of data in adjacent sliding windows, the privacy budget is dynamically allocated which is used to solve the problem of image privacy protection. This method not only satisfies the differential privacy but also has high usability of the published image.

2. Background

2.1. Differential Privacy. Dalenius raised a problem with statistical databases: by accessing the database, no one should be able to get any information about a person [55]. However, due to background knowledge, absolute privacy protection is not possible. Differential privacy sidesteps this issue and turns to relative privacy protection. Any potential privacy breach will be limited to a small multiplier. To be in attention is that serious leaks may occur, but it is not because of whether a particular piece of data exists in the database.

As a carrier of information, a digital image is usually stored and transmitted by a 3D matrix (i.e., a color image can be expressed as R, G, and B, three 2D matrices). To facilitate the data processing, it tries to make normalization treatment to the 3D image matrix and then get the corresponding 2D image gray matrix. Image x can be expressed as a 2D matrix X_{mn}, with m is the number of rows, and N is the number of columns of the matrix.

Formula (1) gives the specific calculation method.

$$X_{mn} = R_{mn} \times 0.299 + G_{mn} \times 0.587 + B_{mn} \times 0.114. \quad (1)$$

Before giving the formal definition of differential privacy, the definition of the neighborhood is given first by combining with X.

Definition 1. Given an image X, the image gray matrix X_{mn} is obtained after normalization, so

$$X|X_{mn} = \begin{bmatrix} x_{11}, x_{12}, & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1}, x_{m1}, & \cdots & x_{mn} \end{bmatrix}; x_{ij} \quad (2)$$

represents the gray value of the corresponding element in the matrix X_{mn}. If there is an X', and there is only one element difference between X and X', that is, $||X - X'| = x_{ij}(1 \le i \le m, 1 \le j \le n)$, then X and X' are said to be adjacent to each other.

Definition 2. A random algorithm M for image data publishing is supposed. Range (M) is the output range of M. if any output s of algorithm M on two two-dimensional matrices X and X' which are adjacent to each other satisfies Equation (3), then algorithm M satisfies ε-differential privacy.

$$\Pr[M(X) \in S] \le \exp(\varepsilon) \times \Pr\left[M\left(X'\right) \in S\right]. \quad (3)$$

In Equation (3), ε is usually a small positive number, which is used to weigh the relationship between privacy and precision. Relatively, if the ε is small, the privacy is higher, and the accuracy is lower, and vice versa. In general, the users select ε by executing a certain privacy policy. Besides, if an algorithm satisfies the ε-differential privacy when the neighbor database differs by one record, it satisfies the $k\varepsilon$-differential privacy when the neighbor database differs at most by K records.

To achieve differential privacy, a certain amount of random noise needs to be added to the query results. Intuitively, its magnitude should cover the maximum impact of a single record on the output. Therefore, the noise level is closely related to the global sensitivity of the corresponding query function.

Definition 3. Q is supposed to any query function and $Q : D \longrightarrow R^d$, the sensitivity of Q is expressed as

$$\Delta Q = \max_{X,X'} \left\| Q(X) - Q\left(X'\right) \right\|_\rho. \quad (4)$$

Laplace mechanism is the most common noise-adding mechanism. For achieving differential privacy, the noise generated by Laplace distribution (the noise distribution satisfies Laplace probability density function $pdf(x \mid \lambda) = (1/2\lambda)$ $e^{-|x|/\lambda}$, $\lambda = \Delta Q/\varepsilon$) is added to disturb the real output.

Theorem 4. *Laplace mechanism: suppose Q is a query sequence of length D. random algorithm m, which takes database as input and outputs the following vectors. It will satisfy ε-differential privacy.*

$$M(D) = Q(D) + < Lap_1\left(\frac{\Delta Q}{\varepsilon}, \cdots, Lap_d\left(\frac{\Delta Q}{\varepsilon} > . \quad (5)\right.\right.$$

What should be noted here is that $\mathrm{Lap}_i(\Delta Q/\varepsilon)$ $(1 \le i \le d)$ is an independent Laplace noise. The magnitude of the noise is proportional to ΔQ and inversely proportional to ε.

Theorem 5. *Exponential mechanism: for any sampling method m under the exponential mechanism, M satisfies ε-differential privacy if it satisfies Equation (6).*

$$M(H, H_i) = \left\{ H_i : \mid \Pr[H_i \in \mathscr{H}] \propto \exp\left(\frac{\varepsilon u(H, H_i)}{2\Delta u}\right) \right\}. \quad (6)$$

The exponential mechanism mainly deals with the nonnumerical output of the sampling algorithm. In the mechanism, $u(H, H_i)$ is the scoring function, Δu is the global sensitivity of scoring function $u(H, H_i)$, and H is the output domain of the algorithm. According to formula (6), the higher the scoring function of H_i, the greater probability of output is selected.

2.2. Data Flow and Sliding Window Model. Differential privacy can ensure that the operation of inserting or deleting a record in a database will not affect the output of any query, thus ensuring that each record's deletion or joining the database will not pose a threat to its privacy. To define differential privacy on a data stream, it is necessary to give the nearest neighbor relationship between two data streams.

Definition 6. For the data stream D and D', if there is at most one record difference between them, then D and D' are neighbors to each other.

Definition 7. For the data stream D and D', they are adjacent to each other. A privacy algorithm A is assumed, if the result s of A on D and $d\wedge$ satisfies Equation (7), then algorithm A satisfies ε-differential privacy.

$$\Pr[A(D) \in S] \le \exp(\varepsilon) \times \Pr\left[A\left(D'\right) \in S\right]. \quad (7)$$

At any time, the sliding window model only needs to consider and process the most recently arrived N data. It can better reflect the characteristics that the importance of data in the data stream gradually decreases with time. Therefore, the sliding window model is usually used in data stream processing. The sliding window model is used to model a data stream D with the length of N. All sliding windows use the fixed size of $|W|$. The value of $|\omega|$ is equal to the number of data contained in the sliding window. The sliding amplitude s is the distance that the current sliding window moves forward compared with the previous one. Generally, the sliding amplitude is 1, which can be adjusted according to the

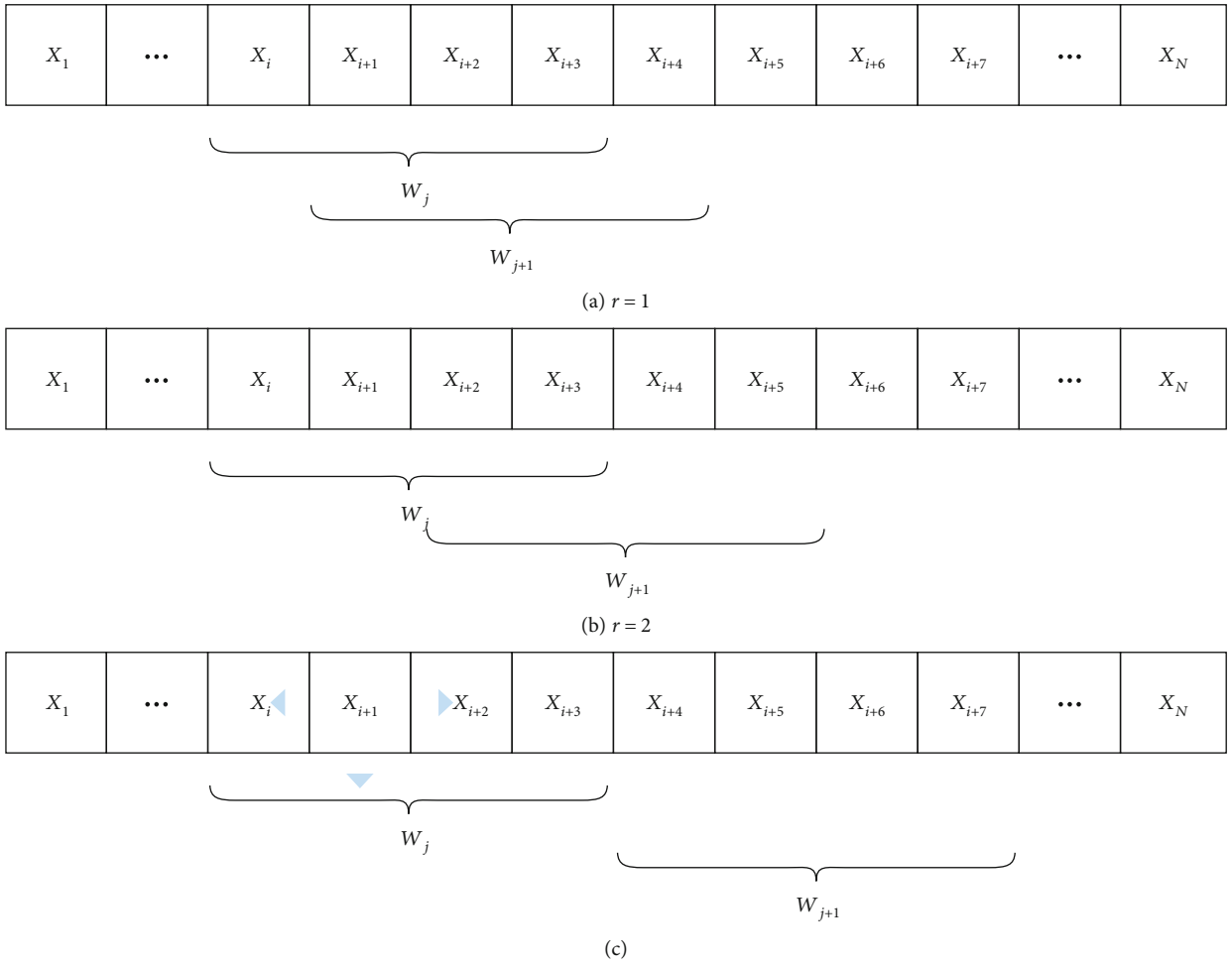


Figure 1: Flow sliding window model.

actual demand. However, to ensure the continuity of the sliding window in the data stream, it is necessary to ensure $1 \le r \le |W|$. The data stream $D = \{x_1, x_2, x_3, \cdots, x_N\}$ and two adjacent sliding windows W_j and W_{j+1}. Assumed $|W| = |W| = 4$ 且 $r = \{1, 2, 4\}$, Figure 1 shows the schematic diagram of adjacent sliding windows under different sliding amplitudes.

3. Methods

Most of the existing differential privacy methods for image publishing use image compression technology to transfer X_{mn}; then, Laplace noise is added to the transformed data. After that, the disturbing data is restored to obtain $X_{mn}{}'$ which includes noise. However, there are two kinds of errors in the process of obtaining $X_{mn}{}'$. One is the noise error LE $(X_{mn}{}')$ which is caused by the Laplace mechanism. The other is the reconstruction error RE $(X_{mn}{}')$ in the process of transformation. As a result, the total error Error $(X_{mn}{}')$ of "X" can be shown.

$$\text{Error}\left(X_{mn}{}'\right) = \text{LE}\left(X_{mn}{}'\right) + \text{RE}\left(X_{mn}{}'\right). \tag{8}$$

The process of converting an image into an image gray matrix can be understood as storing the gray value in each position of the corresponding two-dimensional matrix. As shown in Figure 2, X_{mn} is supposed to give each value a two-dimensional number to represent its position. This paper proposes to reconstruct the original two-dimensional number into a one-dimensional number, transform the image gray matrix into a data stream in the order of the one-dimensional number, and then, use the sliding window model to construct the data stream.

Because the image gray matrix does not have the overall mathematical meaning, namely, the numerical of X_{mn} is no correlation. It can be seen that the transformation of the 2D matrix into a 1D data stream does not destroy the distribution of the original image. Be in attention, the whole conversion process is reversible and lossless, so RE $(X_{m\times n}{}')$ is avoided. In the research of this paper, the total error of $X_{mn}{}'$ is expressed as

$$\text{Error}\left(X_{mn}{}'\right) = LE\left(X_{mn}{}'\right). \tag{9}$$

In this paper, the imaging differential privacy publishing method uses a sliding window model to construct the data

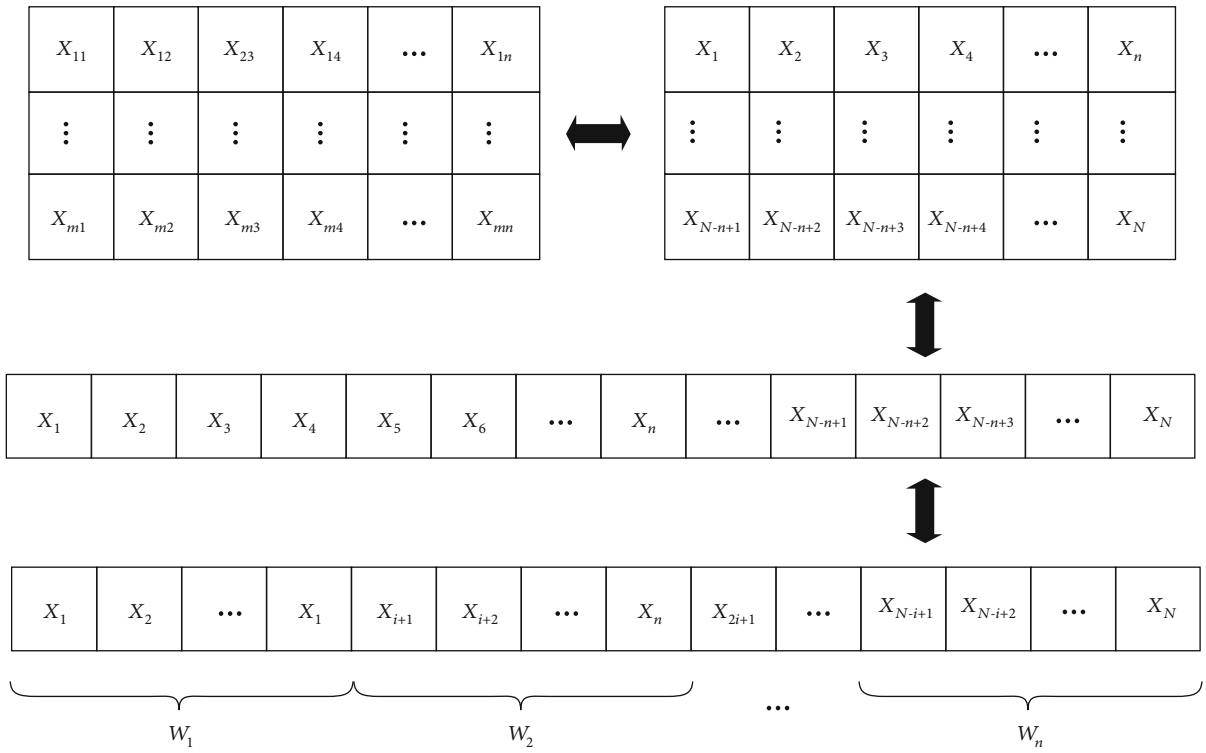


Figure 2: Matrix transform data flow.

stream. Assuming that there is a sliding window W_{j+1} at the moment of $j+1$, and set $r = |W|$. Whether the data in the current sliding window is noisy and published, it needs to be judged according to the preset threshold value. If W_{j+1} and the latest published W'_j (W'_j has been added noise by Laplace mechanism) is less similar than the default threshold, W'_j replace W_{j+1} to be released; otherwise, W_{j+1} is released as W'_{j+1} after it is allocated an appropriate privacy budget and adds noise. In the process of method implementation, the following three principles should be considered:

(1) The data samples contained in any two adjacent sliding windows are all from the image gray matrix. Comparing the similarity between the two samples by numerical difference alone cannot show all the features of the image. Therefore, the measurement method of samples similarity needs to be considered and designed from the image features

(2) Using a sliding window model to construct data flow needs to consider the allocation of the privacy budget. For any W'_j, the smaller privacy budget is allocated, the greater noise value is added, and the higher degree of privacy protection has, but this practice also caused the lower availability of W'_j. Therefore, it is necessary to establish a reasonable privacy budget allocation mechanism on the premise of satisfying ε-differential privacy

(3) The method should satisfy ε-differential privacy and improve the usability of the noisy image

3.1. LAP Algorithm. In this paper, the Laplace algorithm is proposed based on the Laplace mechanism. The method does not change the original data, but directly uses Laplace noise to disturbed the value of X_{mn} and then release $X_{mn}{}'$.

Given the gray matrix X_{mn} of an image, Laplace noise is added to each numerical matrix to obtain $X_{mn}{}' = X_{mn} + \sum_{i=1}^{m\times n} \mathrm{lap}(\Delta Q/\varepsilon)$. According to Theorem 4, the LAP algorithm satisfies the ε-differential privacy, and $\varepsilon = \varepsilon_{\mathrm{sum}}/mn$ is the privacy budget that is allocated to each number of the image gray matrix. $\varepsilon_{\mathrm{sum}}$ is the total privacy budget, and ε is the privacy budget allocated to each numerical. To explain the problem more simply, the size of ε will not be specified under the same conditions.

This paper will continue to describe the error size of the lap algorithm. Since there are only Laplace noises in the algorithm, the sum of squares of errors is obtained as:

$$\begin{aligned}\mathrm{Error}\left(\mathrm{X_{m\times n}}'\right) &= E\left(\sum_{i=1}^{m\times n}\left(x'_i - x_i\right)^2\right)\\ &= E\left(\sum_{i=1}^{m\times n}\left(x_i - x_i + \mathrm{lap}\left(\frac{\Delta Q}{\varepsilon}\right)\right)^2\right)\\ &= 2\mathrm{mn}\left(\frac{\Delta Q}{\varepsilon}\right)^2.\end{aligned} \tag{10}$$

In formula (10), the sensitivity is $\Delta Q = x_{\max} - x_{\min}$. Because the value of X_{mn} comes from the calculation result of the formula (1), so $\Delta Q \le 255$ exists. Besides, it can be seen from the above formula that the main factor determining the

error size in the lap algorithm is the selected image size ($m \times n$). When the selected image size is too large, using the lap algorithm will produce a large amount of noise, which will lead to the image with low usability after adding noise. To improve usability, this paper proposes an image privacy-preserving publishing method based on a sliding window.

3.2. The Measurement Method of Similarity. The image privacy protection publishing method proposed in this paper needs to rely on an accurate similarity measurement method. Different from an ordinary matrix, an image gray matrix not only has mathematical meaning but also includes a color feature, texture feature, and spatial distribution feature of the image itself. The commonly used similarity measurement methods include Manhattan distance, Chebyshev distance, and cosine distance. Due to the existing methods cannot analyze and compare samples from the perspective of images, the results obtained are available, but it will affect the usability of privacy protected images. Therefore, an accurate image data similarity measurement method—EM (exact mechanism)—is proposed.

Suppose W_x and W_y are two adjacent samples obtained by using a sliding window model after the image gray matrix is transformed into a data stream, where $W_x = \{X_1, X_2, X_3, \cdots, X_n\}$, $W_y = \{Y_1, Y_2, Y_3, \cdots, Y_n\}$. Euclidean distance is often used to compare the similarity of data. The formula is:

$$d(X, Y) = \sqrt[2]{\sum_{i=1}^{n} (X_i - Y_i)^2}. \tag{11}$$

Since the data in the sample comes from pixel values, the Euclidean distance is used to measure the similarity of image data, and the results can reflect the difference of color features between different samples. According to formula (11), the similarity formula of the two samples is

$$\theta = d(W_x, W_y). \tag{12}$$

There is an example to illustrate the limitation of θ obtained from Equation (12): there are three sample W_x, W_y, and W_z. Equation (12) is used to calculate θ_1 of W_x and W_y and θ_2 of W_x and W_z. If $\theta_1 < \theta_2$, W_x and W_y can be determined more similar. But actually, it is not like this. Different color feature, texture feature, is not based on pixel feature. It needs statistical calculation in the region containing multiple pixels. In pattern matching, this regional feature has great advantages and will not fail to match due to local bias. Suppose $W_x = \{0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}$, $W_y = \{80,80,80,80,80,80,80,80,80,80\}$, and $W_z = \{0, 0, 0, 0, 0, 0, 0, 0, 0, 255\}$, there is a large global deviation between W_x and W_y, and a minimum local deviation between W_x and W_z (only one data difference between different samples). Equation (12) does not consider the influence of texture features on image matching results.

Jaccard similarity index is used to measure the similarity between two sets. It is defined as the number of elements in the intersection of two sets divided by the number of elements in the union, the formula is

$$\rho(X, Y) = \frac{|X \cap Y|}{|X \cup Y|}. \tag{13}$$

For considering the influence of texture features, the Jaccard similarity index of two samples is used as the optimal value to add to the calculation formula of θ, the formula is

$$\theta = \frac{d(W_x, W_y)}{\rho(W_x, W_y)} + \sigma. \tag{14}$$

In the formula, $d(W_x, W_y)$ is the Euclidean distance between two samples. $\rho(W_x, W_y)$ is the Jaccard similarity index between the two samples. σ appears as a correction value to ensuring the denominator is not 0. Besides, to reduce the influence of $\rho(W_x, W_y)$ to the calculation results of formula (14), the value range of σ should be controlled between 0.8 and 1.

Besides, hamming distance is often used to reflect the difference between two spatial vectors with the same structure and size. Hamming distance is a judgment method by comparing the values of the corresponding positions of two space vectors X and Y in turn. If the value of X_i and Y_i is the same, the result is 0. If it is different, it is 1. The result is to compare the values of all positions of the two space vectors in turn and then accumulate the result 1. The value obtained is the Hamming distance of the two space vectors. The formula of hamming distance is

$$\mu(X, Y) = \sum_{i=1}^{n} X_i \oplus Y_i. \tag{15}$$

The hamming distance can effectively measure the difference of spatial distribution characteristics between two samples, but its calculation result is a positive integer accumulated by the number 1, which cannot be directly used in the calculation of θ. Besides, the calculation results of Hamming distance are limited by the sample size, and the Hamming distance of different size samples is not comparable. To solve this problem, a method combining Hamming distance with a perceptual hash algorithm [55] is proposed to add a disturbance value φ to formula (14). It further enhances the accuracy of θ. The values of φ are as follows:

$$\begin{cases} \dfrac{\theta'(\rho(W_x, W_y) + \sigma)}{d(W_x, W_y)}, \dfrac{\theta'(\rho(W_x, W_y) + \sigma)}{d(W_x, W_y)}, \mu(W_x, W_y) < 0.078n, \\ 1, 0.078n \le \mu(W_x, W_y) \le 0.156n, \\ \dfrac{1}{\lg(n - \mu(W_x, W_y))}, \mu(W_x, W_y) > 0.156n. \end{cases} \tag{16}$$

In the above formula, n is the size of the sample, and θ' is the minimum value given in advance. If $\theta > \theta'$, the two

samples are not similar. If $\theta \le \theta'$, the two samples are similar. To sum up, the EM mechanism can be expressed as

$$\theta = \varphi\left(\frac{d(X, Y)}{\rho(X, Y)} + \sigma\right). \tag{17}$$

3.3. SWP Algorithm. Different from the lap algorithm, SWP (sliding window publication) uses a sliding window model to construct data flow. Through continuous moving of sliding window, it calculates $\theta(2 \le i \le k)$ between W_i and W'_{i-1} by EM mechanism. If $\theta \le \theta', W'_{i-1}$ replaces W_i to release W'_{i-1}; if $\theta > \theta'$, the DA mechanism is used to allocate an appropriate privacy budget to get W'_i, then release W'_i. The implementation process of the SWP algorithm is as follows.

Note that in the SWP algorithm, we use the iterative method to take a dynamic allocation privacy budget mechanism—DA (dynamic allocation) mechanism which can be used for a limited data stream. Different from the commonly used dichotomy privacy budget allocation mechanism, the DA mechanism solves the problem of too fast consumption of privacy budget in a data stream and reduces the impact of noise on the original data to a certain extent.

3.3.1. Dynamic Allocation. In the DA mechanism, the privacy budget for each consumption is assumed to be $\varepsilon_i = \{\varepsilon_1, \varepsilon_2, \varepsilon_3, \cdots, \varepsilon_m\}$, and the number of sliding windows affected by each consumption of privacy budget is $s_i = \{s_1, s_2, s_3, \cdots, s_m\}$, there is $s_1 + s_2 + s_3 + \cdots + s_m = k$, and $s_1 = 1$, K is the total number of sliding windows. Besides, $\varepsilon_i^{\text{left}}$ is the remaining privacy budget after i status is completed, then there is

$$\begin{aligned}
\varepsilon_1 &= \frac{\varepsilon}{k}, \quad \varepsilon_1^{\text{left}} = \frac{\varepsilon - \varepsilon}{k},\\
\varepsilon_2 &= \frac{\varepsilon_1^{\text{left}}}{(k - s_1)}, \quad \varepsilon_2^{\text{left}} = \varepsilon_1^{\text{left}} - \varepsilon_2,\\
\varepsilon_3 &= \frac{\varepsilon_2^{\text{left}}}{(k - s_1 - s_2)}, \quad \varepsilon_3^{\text{left}} = \varepsilon_2^{\text{left}} - \varepsilon_3,\\
\varepsilon_4 &= \frac{\varepsilon_3^{\text{left}}}{(k - s_1 - s_2 - s_3)}, \quad \varepsilon_4^{\text{left}} = \varepsilon_3^{\text{left}} - \varepsilon_4,\\
m-1.\varepsilon_{m-1} &= \frac{\varepsilon_{m-2}^{\text{left}}}{(k - s_1 - s_2 \cdots - s_{m-1})}, \quad \varepsilon_{m-1}^{\text{left}} = \varepsilon_{m-2}^{\text{left}} - \varepsilon_{m-1},\\
m.\varepsilon_m &= \frac{\varepsilon_{m-1}^{\text{left}}}{(k - s_1 - s_2 \cdots - s_m)}, \quad \varepsilon_m^{\text{left}} = \varepsilon_{m-1}^{\text{left}} - \varepsilon_m.
\end{aligned} \tag{18}$$

We can see from the above formula of DA that under any state, ε_i, and $\varepsilon_{i\text{left}}$ the general formula of ε_i and $\varepsilon_{i\text{left}}$ is as follows

$$\varepsilon_i = \frac{\varepsilon_{i-1}^{\text{left}}}{\left(k - \sum_{j=1}^{i-1} s_j\right)}, \tag{19}$$

$$\varepsilon_i^{\text{left}} = \varepsilon_{i-1}^{\text{left}} - \varepsilon_i. \tag{20}$$

As shown in Figure 3, in the implementation process of the DA mechanism, W_1 is added Laplace noise to generate W'_1 and publish it; the privacy budget ε/k is consumed with W'_1 and then publish it. Using EM mechanism to calculate the θ between W'_1 and W_2, if the condition $(\theta <= \theta')$ is satisfied, then W_2 is replaced by W'_1 and released. The above operation is repeated until finding all W_i that meet the conditions. When W_2, W_3, and W_4 are the suitable conditions, all of them are replaced with W'_1. Because the replacement operation does not consume the privacy budget, the total privacy budget consumed by the four sliding windows is ε/k. If θ between W'_1 and W_2 does not satisfy the condition $(\theta > \theta')$, W_5 adds Laplace noise to generate W'_5 and publish it; the privacy budget which cost by W_5 is $\varepsilon(k-1)/k(k-4)$..

Theorem 8. *In SWP, the privacy budget consumed will not exceed ε, that is $\varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \cdots + \varepsilon_m \le \varepsilon$.*

Certificate: according to formula (17), when $k - \sum_{j=1}^{m-1} s_j = 1$, $\varepsilon_{m-1}^{left} = \varepsilon_m$, then $\varepsilon_m^{left} = 0$. In this state, it is expressed as

$$\varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \cdots + \varepsilon_m = \varepsilon, \tag{21}$$

where $k - \sum_{j=1}^{m-1} s_j > 1$, $\varepsilon_{m-1}^{left}/\varepsilon_m > 1$, then $\varepsilon_m^{left} > 0$. In this state, the formula is

$$\varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \cdots + \varepsilon_m < \varepsilon. \tag{22}$$

The proof is complete.

Property 1 sequence combination property: D is assumed as a privacy data set, $A_1, A_2, A_3, \cdots, A_n$ is n random algorithms, and $A_i(1 \le i \le n)$ satisfies ε-differential privacy with $\varepsilon = \sum_{i=1}^{n} \varepsilon_i$.

Theorem 9. *SWP method satisfies ε-differential privacy*

Certificate: It can be seen from Theorem 4:

$$\varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \cdots + \varepsilon_m \le \varepsilon. \tag{23}$$

In the formula, $\varepsilon_i(1 \le i \le m)$ is the budget privacy which is consumed in each state. According to the sequence combination property of difference privacy (property 1), the SWP method satisfies ε-differential privacy. The proof is complete.

Theorem 10. *The error of the SWP method is not greater than the lap algorithm, it is:*

$$Error(SWP) \le Error(Lap). \tag{24}$$

Certificate: the DA used in the SWP method consumes a privacy budget of $\varepsilon_i = \{\varepsilon_1, \varepsilon_2, \varepsilon_3, \cdots, \varepsilon_m\}$. The number of sliding windows affected by each consumption of privacy budget is $s_i = \{s_1, s_2, s_3, \cdots, s_m\}$, and there is $s_1 + s_2 + s_3 + \cdots + s_m = k$. The Laplace mechanism adopted by the lap algorithm consumes a privacy budget ε/k and $\sum_{i=1}^{k} s'_i = k$ $(s'_1 = s'_2 = s'_3 = \cdots = s'_k = 1)$

input : sliding window $W = \{W_1, W_2, W_3, W_4, \cdots, W_k\}$, privacy budget ε, θ'
output : 1. $s = 0$;
2. $W_1' = W_1 + \text{lap}(\varepsilon/k)$;
3. $\varepsilon_{\text{left}} = \varepsilon(k-1)/k$;
4. for $i = 2 + s$ to k
5. use EM mechanism to calculate θ between W_{1+s}' and W_{2+s};
6. If $\theta <= \theta'$
7. $W_i = W_{1+s}'$;
8. $s = s + 1$;
9. else
10. $s = s + 1$;
11. $W_i' = W_i + \text{lap}(\varepsilon_{\text{left}}/(k-s))$;
12. $\varepsilon_{\text{left}} = \varepsilon_{\text{left}} - \varepsilon_{\text{left}}/(k-s)$;
13. $W_{1+s}' = W_i'$;
14. end if
15. end for
16. return $W' = \{W_1', W_2', W_3', W_4', \cdots, W_k'\}$.

Algorithm 1: SWP.

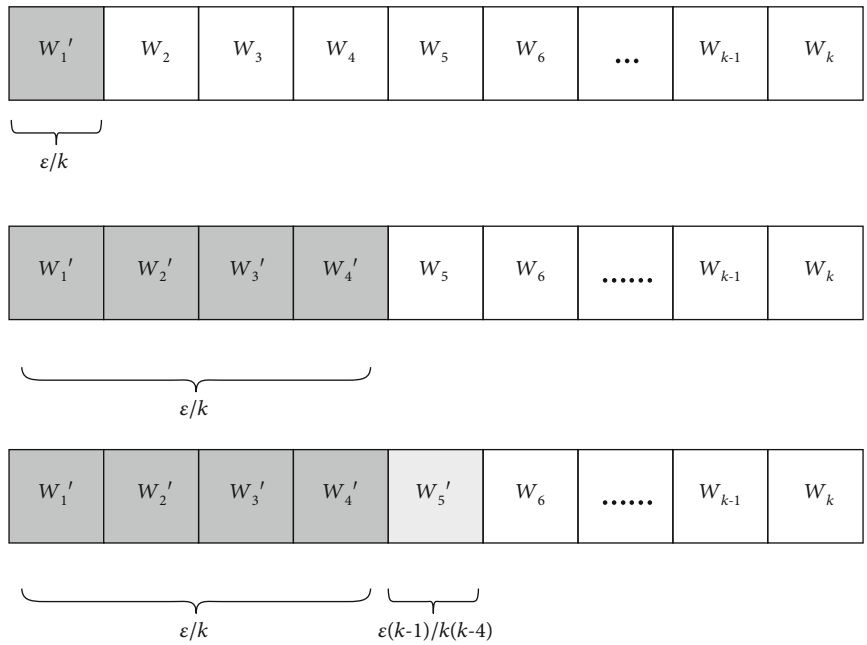


Figure 3: Private budget allocation.

First of all, the maximum error caused by the DA mechanism is calculated. In this case, it knows $s_1 = s_2 = s_3 = \cdots s_m = 1$, means $m = k$, so it is

$$\begin{aligned}\text{Error(SWP)}_{\max} &= \left(\text{lap}\left(\frac{\Delta Q}{\varepsilon_1}\right) + \text{lap}\left(\frac{\Delta Q}{\varepsilon_2}\right) + \text{lap}\left(\frac{\Delta Q}{\varepsilon_3}\right) + \cdots \text{lap}\left(\frac{\Delta Q}{\varepsilon_m}\right)\right)_{\max} \\ &= \sum_{i=1}^{k} \text{lap}\left(\frac{\Delta Q}{\varepsilon_i}\right) = k \times \text{lap}\left(\frac{\Delta Q}{(\varepsilon/k)}\right) = \text{Error(Lap)}.\end{aligned} \tag{25}$$

For other cases, it can be judged according to formula (20). After the completion of state i, the remaining privacy budget of DA is $\varepsilon_i^{\text{left}}$, the remaining privacy budget of the Laplace mechanism is $\acute{\varepsilon}_i^{\text{left}}$. The DA allows $s_i > 1$ and assume the existence of $s_1 = s_2 = s_3 = \cdots s_{i-1} = 1 (1 < i < m)$. From this, it can get

$$\begin{aligned}\varepsilon_i^{\text{left}} - \acute{\varepsilon}_i^{\text{left}} &= \left(\varepsilon - \varepsilon \times \frac{k - (i - 1 + s_i)}{k}\right) - \left(\varepsilon - \varepsilon \times \frac{k-i}{k}\right) \\ &= \varepsilon \times \left(\frac{k-i}{k} - \frac{k-(i-1+s_i)}{k}\right) = \varepsilon \times \frac{s_i - 1}{k}.\end{aligned} \tag{26}$$

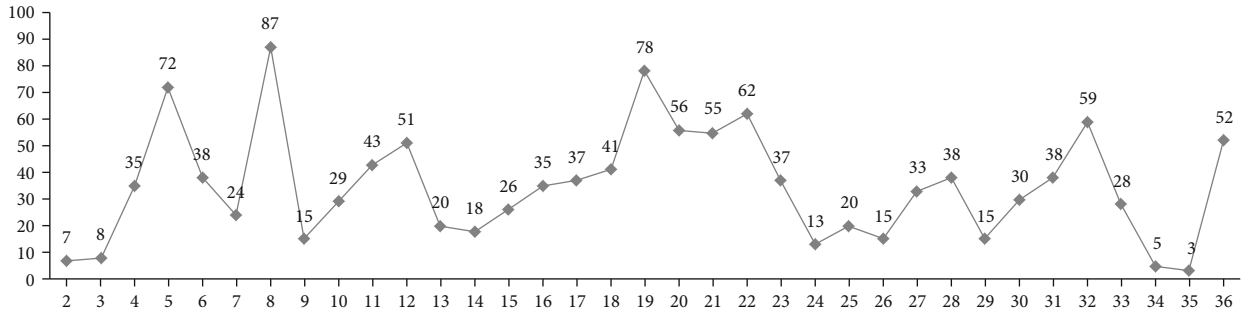


Figure 4: Original sequence.

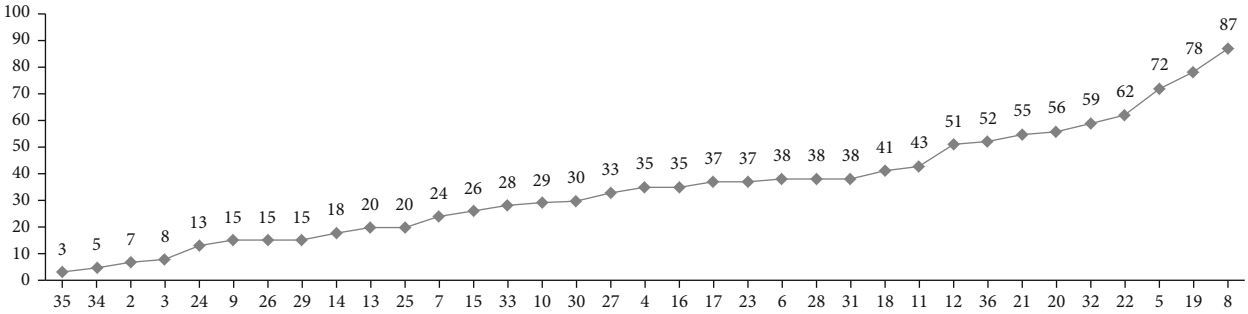


Figure 5: Sort sequence.

Due to $s_i > 1$, so $\varepsilon_i^{\text{left}} - \varepsilon_i^{\text{left}} > 0$, then $(\varepsilon_{i+1}, \varepsilon_{i+2}, \varepsilon_{i+3}, \cdots, \varepsilon_m) > \varepsilon/k$, it is shown that Error(SWP) < Error(Lap).The proof is complete.

3.4. Sort-SWP Algorithm. In the SWP algorithm, the sliding window is set to move forward only in one-dimensional space, so the impact range s_i of W_i' is completely determined by θ'. The verification method is not complicated. Assuming that there is an image x which use the SWP method to protect ε-privacy protection of X (assuming $k = 36$), and θ_i between W_1' and every $W_i (2 \le i \le 36)$ is calculated. Figure 4 shows the distribution of specific data (abscissa is i, the ordinate is θ). When $\theta' = 10$, W_2 and W_3 will be replaced by W_1' to publish. From this, we can see that $s_1 = 2$.

However, by looking at all θ_i in Figure 3, it is found when $\theta' = 10$, it is not only W_2 and W_3 satisfy the condition $\theta \le \theta'$ but also including W_{34} and W_{35}. According to the proof process of Theorem 10, under the same conditions, the larger the value of s_i is, the smaller the $\text{Error}(X')$ is. If the SWP method is executed after sorting θ, assumed $\theta' = 10$, $s_1 = 4$ can be obtained. The sorted result is shown in Figure 5.

However, the direct sorting of data will destroy the ε -differential privacy [56]. Reference [57] proposed to add independent Laplace noise to each subdataset and then use the added noise value to sort. But this method will cause inaccurate sorting results. Reference [58] proposed to use the repeated index mechanism to sample the data and then to sort the data according to the order of sampling. However, the accuracy of the ranking results is too dependent on the allocation of the privacy budget. Therefore, an approximate sorting method—SAS (Simulated Annealing-Sort)—is proposed to satisfy ε-differential privacy. In the SAS algorithm, the sampling technique in exponential mechanism is used to get the initial sorting results, and then Simulated Annealing Algorithm is used to optimize the initial sorting results, so more accurate sorting results can be obtained.

The Simulated Annealing Algorithm comes from the principle of solid annealing, and its core is the choice of solution. Simulated Annealing Algorithm is also a greedy algorithm. Random factors are introduced in its implementation process, which makes the algorithm accept a solution that is worse than the current solution with a certain probability. The annealing process in thermodynamics can be summarized as the state that the body with variable temperature drops slowly and reaches the lowest energy among molecules. Suppose there are n discrete states in the thermodynamic system S, and the energy of the current state is E_{old}, the energy of the next state is E_{new}. According to the Metropolis criterion, the probability P of state transition at temperature T $(1 \le \text{old}, \text{new} \le n)$ is expressed as follows:

$$p = \begin{cases} 1 & E_{\text{new}} < E_{\text{old}}, \\ \exp\left(-\dfrac{E_{\text{new}} - E_{\text{old}}}{\gamma T}\right) & E_{\text{new}} \ge E_{\text{old}}. \end{cases} \tag{27}$$

According to Equation (27), the receiving state is transferred when $E_{\text{new}} < E_{\text{old}}$. It will accept the state transition with probability P when $E_{\text{new}} \ge E_{\text{old}}$. As a positive number which is less than 1, γ is used to control the change rate of temperature T by an iterative method, which also shows that

Input : sliding window $W = \{W_1, W_2, W_3, W_4, \cdots, W_k\}$, Privacy budget ε_1 and ε_2, initial temperature T, Cooling coefficient γ, Termination temperature $T_{\min}$
Output : Approximation of $\tilde{W}$
1. $W_1' = W_1 + \text{lap}(\varepsilon_2)$;
2. Using *EM* mechanism to calculate the similarity between W_1' and all other sliding windows, get $\theta = \{\theta_1, \theta_2, \theta_3, \cdots, \theta_{k-1}\}$;
3. $z = 0$;
4. *while* $(k - z > 1)$
5. The scoring function of exponential mechanism ΔQ is $1/\theta + \sigma$;
6. An unlabeled $\theta_j j = 1 \ldots k - 1$ is selected with probability P $\propto \exp(\varepsilon_1 \Delta Q / 2\Delta u)$ by using the exponential mechanism
7. marked θ_j which is currently selected
8. $z = z + 1$;
9. $\widetilde{\theta}_z = \theta_j$;
10. $S_z = j$;
11. while $(T > T_{\min})$
12. Random select $\widetilde{\theta}_i i = 1 \cdots z - 1$;
13. $dE = \widetilde{\theta}_i - \widetilde{\theta}_{i+1}$;
14. *if*$(dE > = 0)$
15. Change the position of $\widetilde{\theta}_i$ and $\widetilde{\theta}_{i+1}$;
16. Update sorting of data in $\widetilde{\theta}$;
17. Change the position of S_i *and* S_{i+1};
18. *else*
19. *if*$(\exp(dE/T) > random(0, 1))$
20. Change the position of $\widetilde{\theta}_i$ and $\widetilde{\theta}_{i+1}$;
21. Update sorting of data in $\widetilde{\theta}$;
22. Change the position of S_i and S_{i+1};
23. *end if*
24. T = T $*$ γ;
25. Return the sliding window $\tilde{W}$ ={W_1, W_M, …} after sorting, initial position M = {$S_1 + 1, \cdots, S_z + 1$}.

Algorithm 2: SAS.

the higher the temperature is, the greater the value of P is. Conversely, the lower the temperature is, the smaller the value of P will be.

Because the Simulated Annealing Algorithm is probability-based, the solution obtained is not necessarily correct. It is not contrary to the core idea of differential privacy. Based on the above analysis, the implementation details of the SAS algorithm are given.

In the SAS algorithm, the first line is to distribute W_1 to get W_1'. In line 2, using EM mechanism calculates the similarity of W_1' with all other sliding windows and stored in θ. Lines 4-7 use the exponential mechanism to rank θ from small to large. The fifth row is the scoring function of the index mechanism. In the SAS algorithm, we want those sliding window which is more similar to W_1' extracted with a higher probability. σ is a small positive number to ensure that the denominator of the scoring function is not 0. Besides, because deleting or adding a record only affects one count of θ, $\Delta u = 1$. In line 9, the sorted results are stored in $\widetilde{\theta}$. Line 10 is to store the sequence number of the position before sorting θ into S. In line 11-24, the annealing algorithm is used to optimize the sorting results. In line 12-17, $\widetilde{\theta}_i$ is randomly selected in $\widetilde{\theta}$, and the position of $\widetilde{\theta}_i$ and $\widetilde{\theta}_{i+1}$ is changed. Assumed energy change from it is $dE = \widetilde{\theta}_i - \widetilde{\theta}_{i+1}$. If $dE > = 0$, the changed $\widetilde{\theta}$ and S are obtained after accepting the adjustment of this position, and the changed $\widetilde{\theta}$ and S are regarded as the starting point of the next change. Lines 18-23, if $de < 0$, it is accepted as the change with a certain probability and make the corresponding adjustment. Note that when $dE < 0$, $\exp(dE/T)$ ranges from 0 to 1. In line 24, the temperature T is adjusted by using γ (the value of γ is generally set as a positive number slightly less than 1). Since it is an annealing process, it can be seen from formula (12) that the value of T will decrease with the increase of iteration times, and this change will cause the acceptance probability to decrease.

Based on the above contents, a sort-SWP method which is an optimization method based on sorting is proposed. The implementation process of the algorithm is shown as follows.

The sort SWP method consists of three main steps: first, the SAS algorithm is used to sort W to get $\tilde{W}$, where ε_1 is used for exponential mechanism, and ε_2 is used in the Laplace mechanism. To ensure the consistency of the sort SWP method, $\varepsilon_2 = \varepsilon_3/k$ is set. Secondly, the SWP algorithm is used to add noise to $\widetilde{W}$, then gets $\tilde{W}'$; the third (lines 5-11), because the distribution of data in $\tilde{W}'$ destroys the structure of the original image, the position number stored in M is used to restore $\tilde{W}'$ to W'.

Theorem 11. *Sort SWP method satisfies ε-differential privacy*

Input : sliding window $W = \{W_1, W_2, W_3, W_4, \cdots, W_k\}$, Privacy budget ε, θ'
Output : Satisfying differential privacy W'
1. $\varepsilon = \varepsilon_1 + \varepsilon_2 + \varepsilon_3$;
2. $\tilde{W} = \mathrm{SAS}(W, \varepsilon_1, \varepsilon_2)$;
3. $\tilde{W}' = \mathrm{SWP}(\tilde{W}, \varepsilon_3)$;
4. for $i = 1$ to k
5. if $i = 1$
6. $W_1' = \tilde{W}_1'$;
7. Else
8. $m = M_{i-1}$;
9. $W_m' = \tilde{W}_i'$;
10. endif
11. end for
12. return $W' = \{W_1', W_2', W_3', W_4', \cdots, W_k'\}$.

Algorithm 3: Sort-SWP.

Certificate: in the sort SWP method, the total privacy budget is divided into three parts, and there exists $\varepsilon = \varepsilon_1 + \varepsilon_2 + \varepsilon_3$. *Inside,* ε_1 *used exponential machine to select the sliding window* W *and the probability is proportional to* $\exp(\varepsilon_1 \Delta Q / 2\Delta u)$, *it shows as*

$$\Pr[M(X, \Delta Q) = W] = \frac{\exp((\varepsilon_1 \Delta Q(X, W))/2\Delta u)}{\sum_{W' \in O} \exp\left(\left(\varepsilon_1 \Delta Q\left(X, W'\right)\right)/2\Delta u\right)}. \tag{28}$$

Given X and its adjacent X', for any value of W ($W \in O$), it can be seen from formula (28)

$$\begin{aligned}
&\frac{\Pr[M(X, \Delta Q) = W]}{\Pr\left[M\left(X', \Delta Q\right) = W\right]} \\
&= \frac{\exp(\varepsilon_1 \Delta Q(X, W)/2\Delta u) / \sum_{W' \in O} \exp\left(\varepsilon_1 \Delta Q\left(X, W'\right)/2\Delta u\right)}{\exp\left(\varepsilon_1 \Delta Q\left(X', W\right)/2\Delta u\right) / \sum_{W' \in O} \exp\left(\varepsilon_1 \Delta Q\left(X', W'\right)/2\Delta u\right)} \\
&= \left(\frac{\exp(\varepsilon_1 \Delta Q(X, W)/2\Delta u)}{\exp\left(\varepsilon_1 \Delta Q\left(X', W\right)/2\Delta u\right)}\right) \times \left(\frac{\sum_{W' \in O} \exp\left(\varepsilon_1 \Delta Q\left(X', W'\right)/2\Delta u\right)}{\sum_{W' \in O} \exp\left(\varepsilon_1 \Delta Q\left(X, W'\right)/2\Delta u\right)}\right) \\
&= \exp\left(\frac{\varepsilon_1\left(\varepsilon_1 \Delta Q(X, W) - \varepsilon_1 \Delta Q\left(X', W\right)\right)}{2\Delta u}\right) \\
&\quad \times \left(\frac{\sum_{W' \in O} \exp\left(\varepsilon_1 \Delta Q\left(X', W'\right)/2\Delta u\right)}{\sum_{W' \in O} \exp\left(\varepsilon_1 \Delta Q\left(X, W'\right)/2\Delta u\right)}\right) \le \exp\left(\frac{\varepsilon_1}{2}\right) \\
&\quad \times \left(\frac{\sum_{W' \in O} \exp(\varepsilon_1/2) \times \exp\left(\varepsilon_1 \Delta Q\left(X, W'\right)/2\Delta u\right)}{\sum_{W' \in O} \exp\left(\varepsilon_1 \Delta Q\left(X, W'\right)/2\Delta u\right)}\right) \\
&\le \exp\left(\frac{\varepsilon_1}{2}\right) \times \exp\left(\frac{\varepsilon_1}{2}\right) \times \left(\frac{\sum_{W' \in O} \exp\left(\varepsilon_1 \Delta Q\left(X, W'\right)/2\Delta u\right)}{\sum_{W' \in O} \exp\left(\varepsilon_1 \Delta Q\left(X, W'\right)/\varepsilon_1 \Delta Q\left(X, W'\right)\right)}\right) \\
&= \exp(\varepsilon_1).
\end{aligned} \tag{29}$$

It can be seen that the process of selecting W by using the exponential mechanism satisfies ε_1-differential privacy in Sort-SWP. Besides, according to Theorem 4, the Sort-SWP method satisfies ε_2-differential privacy. According to the proof process of Theorem 9, it can be concluded that the Sort-SWP method satisfies ε_3-differential privacy. Therefore, the whole process of the Sort-SWP method satisfies the ε -differential privacy. The proof is complete.

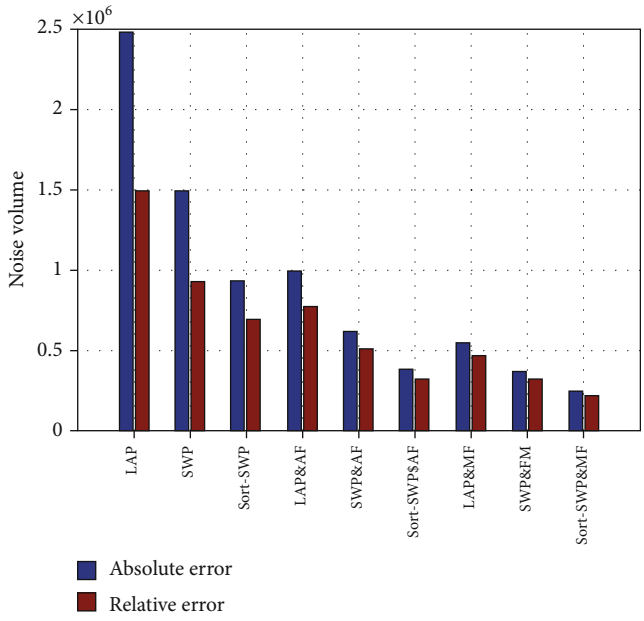
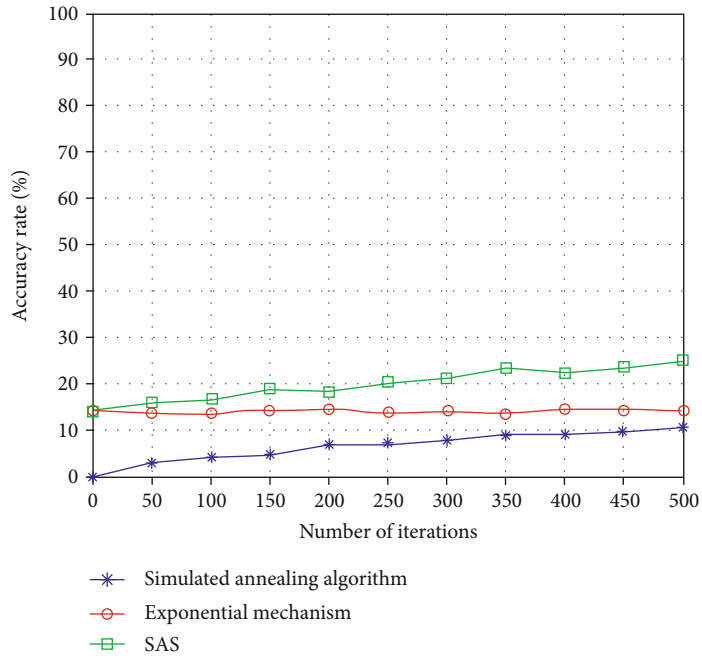


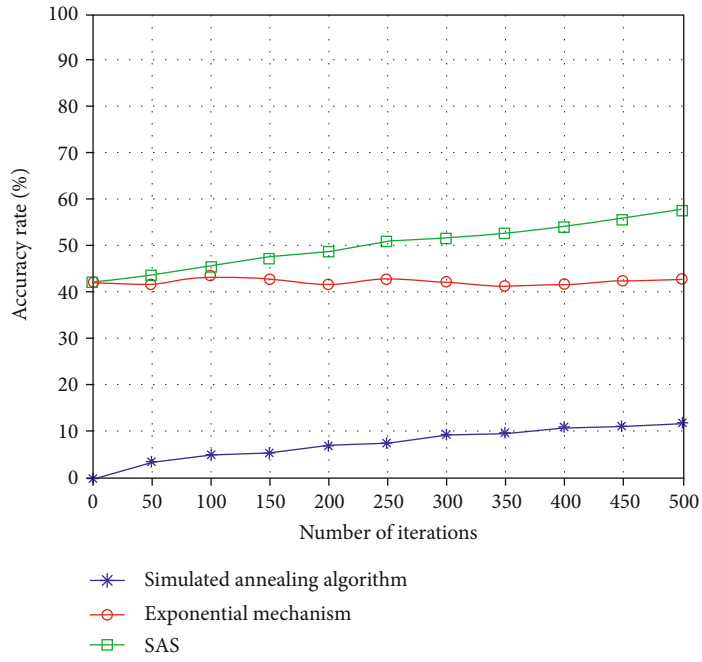
Figure 6: Noise statistics.

4. Experiment and Result Analysis

4.1. Sorting Method Experiment. Direct sorting will destroy the ε-differential privacy. Therefore, this paper adopts an approximate sorting method—SAS algorithm—which satisfies the ε-differential privacy. During the test, a one-dimensional array which is containing 57 random numbers is used as the experimental data set, and three sorting methods, namely, simulated annealing algorithm, exponential mechanism, and *SAS* algorithm, are selected for comparison. Besides, to verify the impact of the privacy budget on

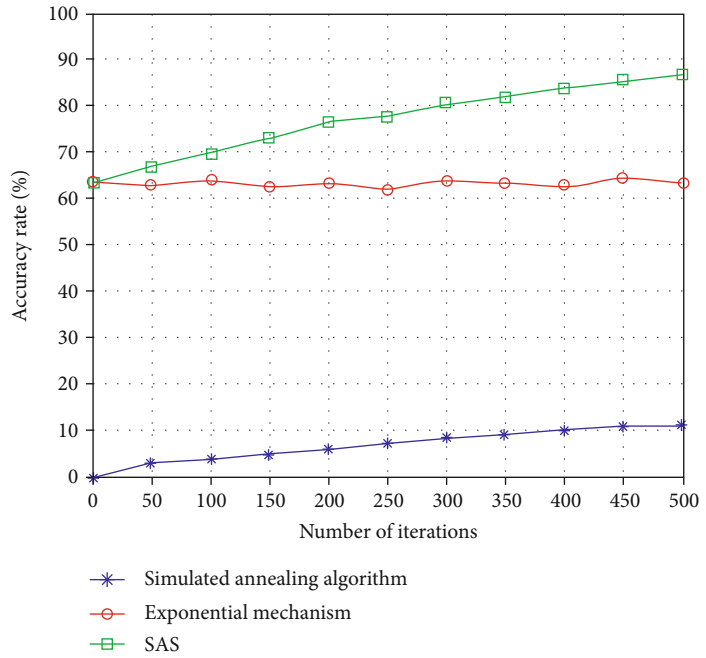


(a) $\varepsilon = 0.4$

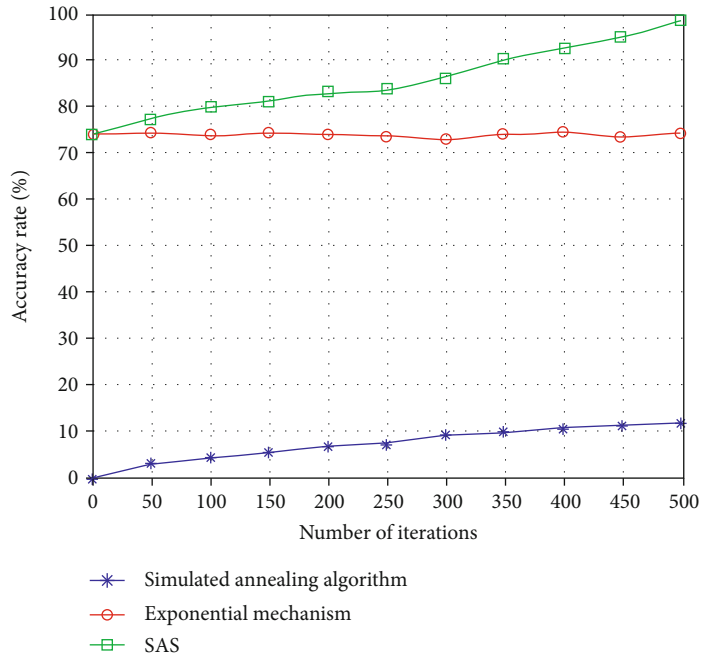


(b) $\varepsilon = 0.5$

Figure 7: Continued.



(c) $\varepsilon = \ln 2$



(d) $\varepsilon = 1$

Figure 7: Sorting result test.

different ranking results, ε is taken as 0.4, 0.5, ln2, and 1, respectively. The results are shown in Figure 6.

In Figure 7, the abscissa represents the number of times the algorithm is executed, and the ordinate represents the accuracy of the sorting. The following three conclusions can be drawn from the experimental results: (1) the sorting results of the simulated annealing algorithm are only related to the execution times of the algorithm and not relevant to the value of ε; (2) the ranking results of the exponential mechanism are only related to the value of ε, and the higher the value of ε is, the higher the sorting accuracy is. It is independent of the execution times of the algorithm; (3) the sorting results of the SAS algorithm are related to the value of ε and also related to the execution times of the algorithm, and they are proportional.

4.2. Simulation Experiment. In different kinds of images, the sensitive information of face photos is the most representative. Therefore, to verify the usability of the algorithm, it selects a front face photo $(190 * 160)$ to complete the

(a) Original image (b) LAP

(c) SWP (d) Sort-SWP

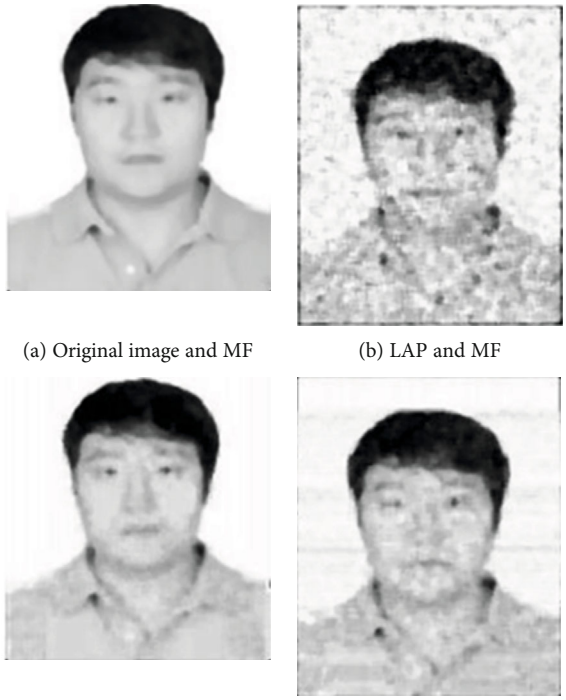
Figure 8: Experimental results.

(a) Original image and AF (b) LAP and AF

(c) SWP and AF (d) Sort-SWP and AF

Figure 9: Experimental results graph and mean filtering.

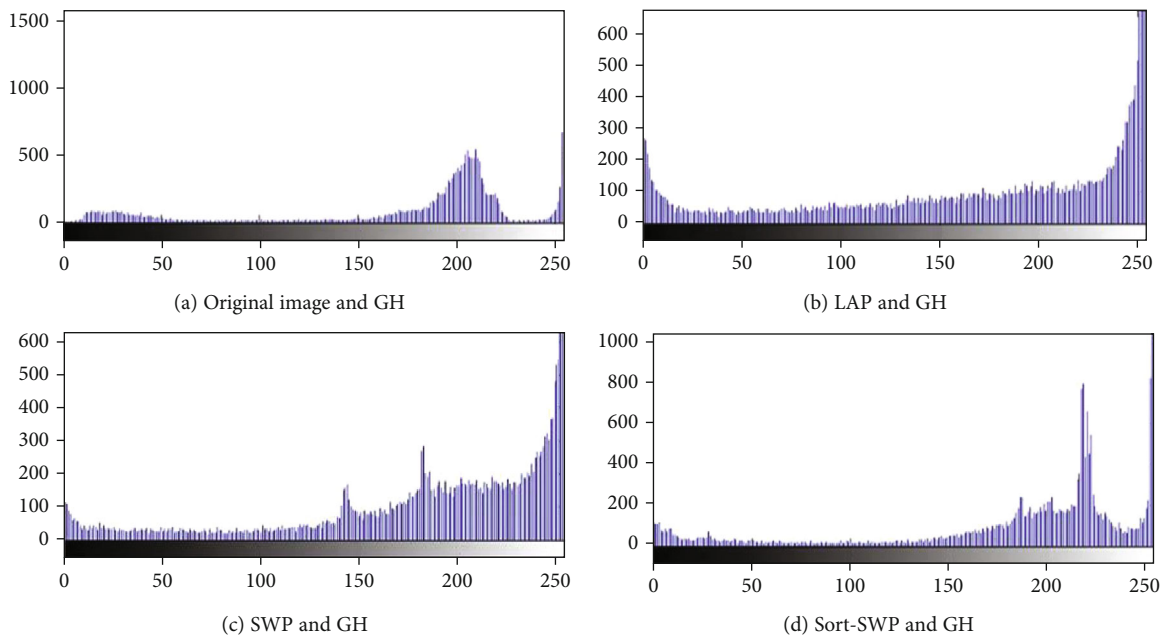
corresponding experiment in this paper. The experimental results of different algorithms under the same conditions are shown in Figure 8. Figure 8(a) is the original image; Figure 8(b) is the resulting graph obtained by directly adding Laplace noise to the original image; Figure 8(c) is the resulting graph obtained by using the SWP algorithm; Figure 8(d) is the resulting graph obtained by using sort SWP algorithm. Compared with the LAP algorithm, the



(a) Original image and MF (b) LAP and MF

(c) SWP and MF (d) Sort-SWP and MF

Figure 10: Experimental result graph and median filter.



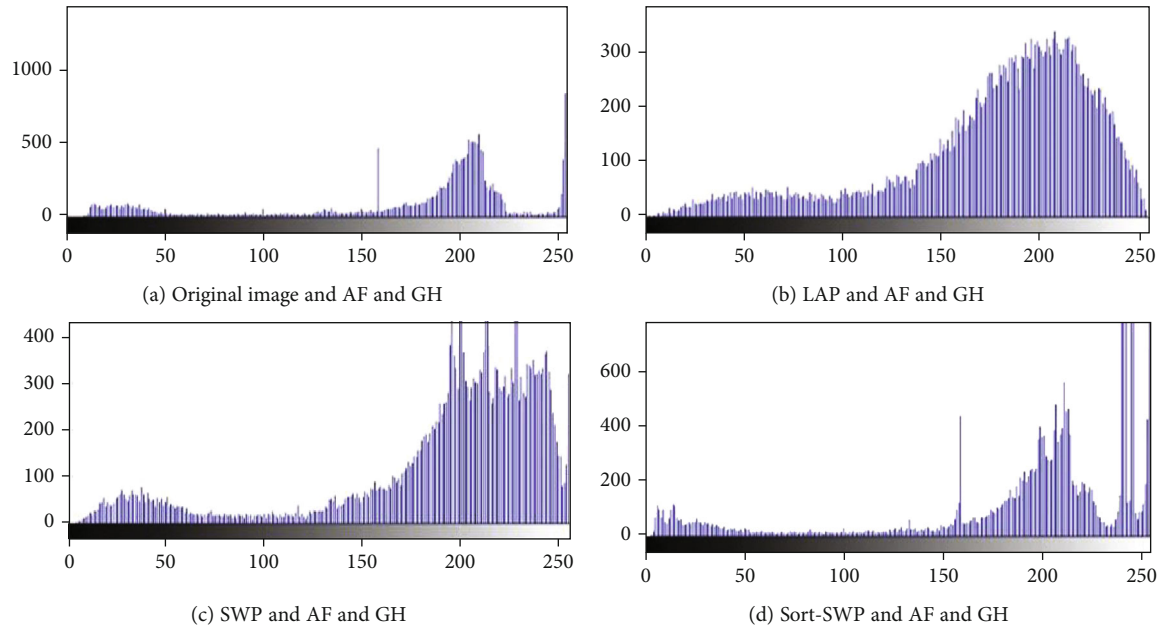
(a) Original image and GH (b) LAP and GH

(c) SWP and GH (d) Sort-SWP and GH

Figure 11: Gray histogram of experimental image.

SWP algorithm and sort SWP algorithm have better display effect.

The core of this paper is to use the Laplace mechanism in differential privacy protection to add noise to the image and to protect the sensitive information in the image. Therefore, to destroy the effect of privacy protection, it is a common method to eliminate or reduce the impact of image noise. For the methods of eliminating image noise, the representative methods are average filtering or median filtering. In Figures 9 and 10, the mean filter and median filter are used to interfere with the experimental results in Figure 8. The purpose of this is to evaluate the degree of privacy protection and the usability of the protected images after noise elimination.



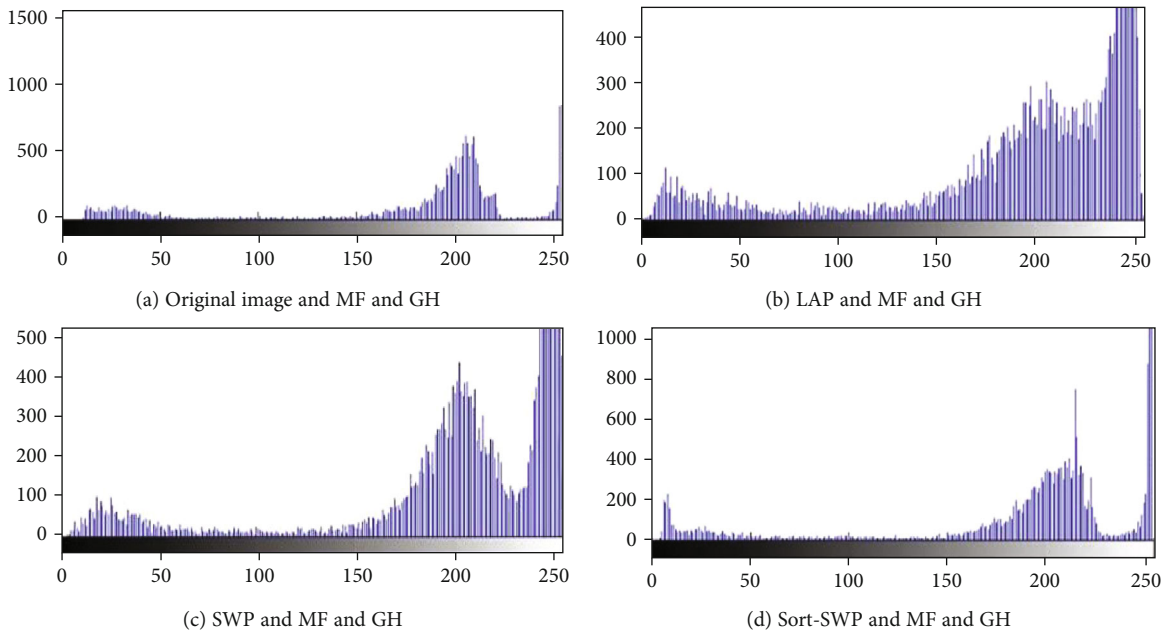
(a) Original image and AF and GH

(b) LAP and AF and GH

(c) SWP and AF and GH

(d) Sort-SWP and AF and GH

Figure 12: Experimental image gray histogram and average filtering.



(a) Original image and MF and GH

(b) LAP and MF and GH

(c) SWP and MF and GH

(d) Sort-SWP and MF and GH

Figure 13: Experimental image gray histogram and median filter.

Now, we have given a preliminary conclusion that the SWP algorithm and sort SWP algorithm are better than the LAF method. However, this conclusion is only obtained by observing the experimental results, so it may be one-sided or inaccurate. To further verify whether the conclusion is correct, the previous 12 images are transformed into a gray histogram for further comparison. Gray histogram reflects the relationship between the frequency of gray level pixels and gray level in an image, although the gray histogram cannot reflect the specific distribution of image pixels but reflect the impact of Laplace noise on the image as a statistical result.

It can be seen from Figures 11–13, under the same conditions, the histogram distribution of the SWP algorithm and lap algorithm is closer, but it is quite different from the distribution of pixels in the original image. The distribution of pixels in the histogram of the result of the sort SWP algorithm is consistent with the original image. Therefore, it can be considered that the noise interference caused by the sort SWP algorithm is the least.

In the papers related to differential privacy protection, the noise size is an important evaluation index to measure the advantages and disadvantages of the algorithm. Under

Table 1: ORL and precision.

Precision	LAP	SWP	Sort-SWP	LAP and AF	SWP and AF	Sort-SWP and AF	LAP and MF	SWP and MF	Sort-SWP and MF
$\varepsilon = 1$	6.52%	39.29%	53.86%	9.27%	47.09%	60.55%	14.95%	49.01%	65.39%
$\varepsilon = 2$	8.83%	43.45%	69.26%	15.86%	56.39%	72.06%	25.37%	57.61%	75.41%
$\varepsilon = 3$	11.37%	48.29%	71.66%	22.00%	63.39%	79.27%	36.61%	72.18%	83.68%
$\varepsilon = 4$	13.96%	55.18%	75.19%	27.71%	70.43%	82.51%	44.36%	80.97%	87.55%
$\varepsilon = 5$	16.67%	62.72%	82.83%	33.08%	76.36%	86.19%	51.98%	88.70%	91.33%

Table 2: ORL and recall.

Recall	LAP	SWP	Sort-SWP	LAP and AF	SWP and AF	Sort-SWP and AF	LAP and MF	SWP and MF	Sort-SWP and MF
$\varepsilon = 1$	71.32%	77.96%	83.58%	72.35%	81.63%	84.57%	80.12%	83.47%	88.14%
$\varepsilon = 2$	73.56%	75.68%	81.83%	75.52%	81.26%	83.67%	78.68%	79.85%	80.33%
$\varepsilon = 3$	75.79%	83.81%	85.25%	77.53%	84.36%	85.23%	81.18%	86.63%	84.54%
$\varepsilon = 4$	72.32%	74.22%	78.58%	73.52%	78.59%	82.18%	75.36%	81.23%	87.10%
$\varepsilon = 5$	77.91%	76.23%	80.12%	78.07%	80.11%	82.51%	80.12%	81.15%	83.94%

Table 3: ORL and F1-score.

F1-sroce	LAP	SWP	Sort-SWP	LAP and AF	SWP and AF	Sort-SWP and AF	LAP and MF	SWP and MF	Sort-SWP and MF
$\varepsilon = 1$	11.95%	52.25%	65.51%	16.43%	59.73%	70.57%	25.20%	61.76%	75.08%
$\varepsilon = 2$	15.77%	55.21%	75.02%	26.21%	66.58%	77.43%	38.37%	66.93%	77.79%
$\varepsilon = 3$	19.77%	61.27%	77.87%	34.27%	72.39%	82.14%	50.46%	78.75%	84.11%
$\varepsilon = 4$	23.40%	63.30%	76.85%	40.25%	74.29%	82.34%	55.85%	81.10%	87.32%
$\varepsilon = 5$	27.46%	68.82%	81.45%	46.47%	78.19%	84.31%	63.05%	84.76%	87.48%

Table 4: YALE and precision.

Precision	LAP	SWP	Sort-SWP	LAP and AF	SWP and AF	Sort-SWP and AF	LAP and MF	SWP and MF	Sort-SWP and MF
$\varepsilon = 1$	4.12%	27.33%	44.52%	7.89%	41.25%	53.91%	10.18%	44.74%	56.10%
$\varepsilon = 2$	6.77%	31.56%	57.38%	12.51%	48.98%	60.47%	18.23%	53.82%	64.58%
$\varepsilon = 3$	9.85%	39.41%	64.57%	18.01%	57.24%	69.13%	29.76%	59.71%	71.35%
$\varepsilon = 4$	11.26%	46.55%	72.31%	23.80%	65.50%	75.25%	37.61%	66.07%	78.69%
$\varepsilon = 5$	13.01%	59.32%	77.01%	27.12%	72.05%	83.27%	46.07%	75.15%	86.38%

the same conditions, less noise represents higher availability. Of course, as a special information carrier, image noise is only one of the evaluation indicators.

If the pixel value of the gray-scale image is regarded as a value, the value after adding Laplace noise will change with the change of ε. The smaller the ε is, the greater the noise is, the higher the degree of privacy protection is, and there is no limit to the value. However, in the experiment, the range of the pixel value of the gray image is limited between 0 and 255, so even if the obtained value is beyond this range under the influence of noise, it will be normalized. That means all values greater than 255 will be changed to 255, and all values less than 0 will be changed to 0. Therefore, the error of the image after privacy protection is divided into absolute error and relative error. For example, in an image x, there is a pixel $x_{mn} = 200$. If $x'_{mn} = 280$ after adding noise, the absolute error value is 80, and the relative error value is 55.

In Figure 6, the absolute error and relative error value of the experimental result graph under the influence of different algorithms are given. The conclusion is that the error caused by the three algorithms is in line with the previous expectations, and compared with the LAP algorithm and SWP algorithm, the sort-SWP algorithm will cause the smallest error. Also, the median filter is slightly better than the average filter for Laplace noise.

4.3. Result Analysis. To verify the feasibility of the algorithm, the ORL face database, and YALE face database are used as

Table 5: YALE and recall.

Recall	LAP	SWP	Sort-SWP	LAP and AF	SWP and AF	Sort-SWP and AF	LAP and MF	SWP and MF	Sort-SWP and MF
$\varepsilon = 1$	77.34%	82.51%	84.18%	79.58%	84.83%	83.05%	73.62%	82.49%	85.26%
$\varepsilon = 2$	78.12%	78.74%	82.47%	77.41%	80.24%	84.18%	76.14%	80.14%	86.14%
$\varepsilon = 3$	80.34%	83.56%	87.51%	78.87%	81.07%	86.91%	79.35%	81.61%	85.36%
$\varepsilon = 4$	79.24%	81.54%	80.63%	76.15%	77.69%	85.43%	80.45%	78.34%	84.81%
$\varepsilon = 5$	75.39%	82.37%	85.66%	78.56%	82.48%	88.75%	79.51%	81.95%	87.57%

Table 6: YALE and F1-score.

F1-sroce	LAP	SWP	Sort-SWP	LAP and AF	SWP and AF	Sort-SWP and AF	LAP and MF	SWP and MF	Sort-SWP and MF
$\varepsilon = 1$	7.82%	41.06%	58.24%	14.36%	55.51%	65.38%	17.89%	58.01%	67.67%
$\varepsilon = 2$	12.46%	45.06%	67.67%	21.54%	60.83%	70.38%	29.42%	64.39%	73.82%
$\varepsilon = 3$	17.55%	53.56%	74.31%	29.32%	67.10%	77.01%	43.29%	68.96%	77.73%
$\varepsilon = 4$	19.72%	59.27%	76.24%	36.27%	71.08%	80.02%	51.26%	71.68%	81.64%
$\varepsilon = 5$	22.19%	68.97%	81.11%	40.32%	76.91%	85.92%	58.34%	78.40%	86.97%

experimental data sets, and the experimental environment is Intel® Corei9-9900K CPU@ 3.60 GHz,32 G memories, GTX 21080TI GPU, Windows 10 operating system. A face recognition method based on an improved AlexNet convolution neural network is adopted in the experiment. Compared with the AlexNet model, this method has a simpler network structure and fewer parameters, which can save a lot of model training time and then make a fast prediction. When using the ORL face database for the experiment, 5 images of each person are used as a training set, and other images are used as a test set. The sliding window length is 56, $k = 184$. When using the YALE face database for experiments, random 5 images of each person are also used as the training set and the rest as the test set. The sliding window length is 160, $k = 486$. Privacy budget ε is 1, 2, 3, 4, and 5. The test items are accuracy rate, recall rate, and F1 score. The experimental results are shown in Tables 1–6.

5. Conclusions

To solve the privacy protection problem of face image publishing, this paper proposes to use the Laplace mechanism of differential privacy to add noise in the image, so that this should protect the sensitive information in the face image. Compared with directly adding Laplace noise to the image, the SWP algorithm and sort SWP algorithm can effectively reduce the impact of noise on the protected image. Especially sort SWP algorithm, which uses the SAS algorithm to sort and then execute the SWP algorithm, is proposed to further improve the usability of the protected face image. From the results of several test experiments, compared with the lap algorithm (ORL, Yale), the absolute error of SWP algorithm decreased by 40.81%, relative error decreased by 38.26%, accuracy increased by 35.01%, recall rate increased by 3.53%, and F1-sroce increased by 39.07%; absolute error of sort-SWP algorithm decreased by 64.12%, relative error decreased by 51.70%, accuracy increased by 56.63%, recall rate increased by 6.85%, and F1-sroce increased by 55.62%. Besides, in the test of SAS algorithm, compared with the exponential mechanism ranking method (the accuracy of simulated annealing algorithm is only related to the number of iterations), the average ranking accuracy of SAS algorithm is improved by 17.63%.

The differential privacy protection method proposed in this paper is implemented by global noise adding. However, most of the sensitive information contained in a face image exists in some specific areas (such as eyes, mouth, nose, and other organs or feature points). The next research direction will be devoted to accurately finding the region of sensitive information and adding noise for those local regions. At the same time, the research will try to further reduce the impact of noise and improve the usability of face image after privacy protection.

Data Availability

The raw/processed data required to reproduce these findings cannot be shared at this time as the data also forms part of an ongoing study.

Conflicts of Interest

The authors declare that they have no competing interests.

Acknowledgments

The authors thank the project of the National Natural Science Foundation of China (Nos. 61672179, 61370083, and 61402126), Natural Science Foundation of Heilongjiang Province (No. F2015030), and Fundamental Research Funds in Heilongjiang Provincial Education Department (Nos. 135109247, 135109243, 12541872, and 135209239).

References

[1] K. Wang, B. C. M. Fung, and P. S. Yu, "Handicapping attackers confidence; an alternative to *k*-anonymization," *Knowledge and Information Systems*, vol. 11, no. 3, pp. 345–368, 2007.

[2] B. C. M. Fung, K. Wang, and P. S. Yu, "Anonymizing classification data for privacy preservation," *IEEE Trans on Knowledge and Data Engineering*, vol. 19, no. 5, pp. 711–725, 2007.

[3] X. Xiao and T. Y. Anatomy, "Simple and effective privacy preservation," in *Proc of the 32nd IntConf on Very Large Data Bases*, pp. 139–150, New York, 2006.

[4] T. Li, N. Li, J. Zhang, and I. Molloy, "Slicing: a new approach for privacy preserving data publishing," *IEEE Trans on Knowledge and Data Engineering*, vol. 24, no. 3, pp. 561–574, 2012.

[5] M. Terrovitis, N. Mamoulis, J. Liagouris, and S. Skiadopoulos, "Privacy preservation by diSASsociation," *Proceedings of the VLDB Endowment*, vol. 5, no. 10, pp. 944–955, 2012.

[6] L. Sweeney, "k-Anonymity: a model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 5, pp. 557–570, 2012.

[7] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "l-Diversity: privacy beyond k-anonymity," in *22nd International Conference on Data Engineering (ICDE'06)*, pp. 1–12, Atlanta, GA, USA, 2006.

[8] N. Li, T. Li, and S. Venkatasubramanian, "t-Closeness: privacy beyond k-anonymity and l-diversity," in *2007 IEEE 23rd International Conference on Data Engineering*, pp. 106–115, Istanbul, Turkey, April 2007.

[9] X. K. Xiao and Y. F. Tao, "Towards privacy-preserving republication of dynamic datasets," in *IEEE 51th Annual Symposium on Foundations of Computer Science*, pp. 619–642, Las Vegas, NV,USA, 2010.

[10] J. Xu, W. Wang, J. Pei, X. Wang, B. Shi, and A. W.-C. Fu, "Utility-based anonymization using local recoding," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '06*, pp. 785–790, Philadelphia, PA, USA, 2006.

[11] J. Zhang, G. Cormode, C. M. Procopiuc, D. Srivastava, and X. Xiao, "Private release of graph statistics using ladder functions," in *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data - SIGMOD '15*, pp. 731–745, Melbourne, Victoria, Australia, 2015.

[12] Y. Chen, A. Machanavajjhala, M. Hay, and G. Miklau, "PeGaSus: data-adaptive differentially private stream processing," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1375–1388, Dallas, TX, USA, October 2017.

[13] O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game," in *Proceedings of the nineteenth annual ACM conference on Theory of computing - STOC '87*, pp. 218–229, New York, USA, 1987.

[14] C. Gentry, "Fullyhomomorphic encryption using ideal lattices," *Stochastics*, vol. 9, p. 169, 2009.

[15] C. Liu, J. Yang, and J. Wu, "Web intrusion detection system combined with feature analysis and SVM optimization," *EURASIP Journal on Wireless Communications and Networking*, vol. 2020, no. 1, Article ID 33, 2020.

[16] S. Ramezanian, T. Meskanen, M. Naderpour, V. Junnila, and V. Niemi, "Private membership test protocol with low communication complexity," *Digital Communications and Networks*, vol. 6, no. 3, pp. 321–332, 2020.

[17] V. Shukla, A. Chaturvedi, and N. Srivastava, "A secure stop and wait communication protocol for disturbed networks," *Wireless Personal Communications*, vol. 110, no. 2, pp. 861–872, 2020.

[18] C. Thammarat and W. Kurutach, "A lightweight and secure NFC-base mobile payment protocol ensuring fair exchange based on a hybrid encryption algorithm with formal verification," *International Journal of Communication Systems*, vol. 32, no. 12, article e3991, 2019.

[19] M. Anbarasan, S. Prakash, A. Antonidoss, and M. Anand, "Improved encryption protocol for secure communication in trusted MANETs against denial of service attacks," *Multimedia Tools and Applications*, vol. 79, no. 13-14, pp. 8929–8949, 2020.

[20] B. Liu, B. Zhan, C. Zhang, and L. Yang, "Research on visual control system of inverted pendulum based on pixel displacement," *Journal of Physics: Conference Series*, vol. 1550, article 062006, 2020.

[21] PixArt Imaging, "Patent issued for high accuracy displacement detection system with offset pixel array (USPTO 10, 609, 314)," pp. 8821–8835, 2020.

[22] A. Grigoriev, E. Danilova, V. Trusov, M. Miheev, and M. Uhanova, "Modelling of measurement error for vibrational displacement based on the blurring analysis of a round mark image," *Applied Computer Systems*, vol. 23, no. 1, pp. 69–74, 2018.

[23] G. Ye and X. Huang, "An image encryption algorithm based on autoblocking and electrocardiography," *IEEE Multimedia*, vol. 23, no. 2, pp. 64–71, 2016.

[24] L. Yuan, P. Korshunov, and T. Ebrahimi, "Privacy-preserving photo sharing based on a secure JPEG," in *2015 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 185–190, Hong Kong, China, April 2015.

[25] Z. Moghaddasi, H. A. Jalab, and R. M. Noor, "Image splicing forgery detection based on low-dimensional singular value decomposition of discrete cosine transform coefficients," *Neural Computing and Applications*, vol. 31, no. 11, pp. 7867–7877, 2019.

[26] L. Yuan and T. Ebahimi, "Image privacy protection with secure JPEG transmorphing," *IEEE Transactions on Signal Processing*, vol. 11, no. 9, pp. 1031–1038, 2017.

[27] M. Sundararajan, M. Veerappan, and S. Anbazhagan, "Partial image encryption based on using discrete cosine transform coefficients and lightweight stream algorithm," *Journal of Computational and Theoretical Nanoscience*, vol. 16, no. 4, pp. 1573–1576, 2019.

[28] L. He, Y. Wang, and Z. Xiang, "Wavelet frame-based image restoration using sparsity, nonlocal, and support prior of frame coefficients," *The Visual Computer*, vol. 35, no. 2, pp. 151–174, 2019.

[29] C. Dwork, "Differential privacy," in *Proceedings of the 33rd International Colloquium on Automata Languages and Programming*, pp. 1–12, Berlin, 2006.

[30] C. Dwork, "Differential privacy: a survey of results," in *Proceedings of the 5th International Conference on Theory and Applications of Models of Computation*, pp. 1–19, Xi'an, China, 2008.

[31] C. Dwork, "The differential privacy frontier (extended abstract)," in *Proceedings of the 6th Theory of Cryptography Conference*, pp. 496–502, San Francisco.CA.UAS., 2009.

[32] C. Dwork and J. Lei, "Differential privacy and robust statistics," in *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, pp. 371–380, Bethesda, MD, USA, 2009.

[33] C. Dwork, "Differential privacy in new settings," in *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms*, Austin, Texas, USA, January 2010.

[34] C. Dwork, "The promise of differential privacy: a tutorial on algorithmic techniques," in *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, Palm Springs, CA, USA, October 2011.

[35] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Proceedings of the 3rd Theory of Cryptography Conference*, pp. 265–284, New York, NY, USA, 2006.

[36] F. McSherry and K. Talwar, "Mechanism design via differential privacy," in *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, pp. 94–103, Providence, Rhode Island, USA, October 2007.

[37] F. McSherry, "Privacy integrated queries," *Communications of the ACM*, vol. 53, no. 9, pp. 89–97, 2010.

[38] J. Xu, Z. Zhang, X. Xiao, Y. Yang, and G. Yu, "Differentially private histogram publication," in *2012 IEEE 28th International Conference on Data Engineering*, pp. 32–43, Washington, DC, USA, April 2012.

[39] X. Liu and S. Li, *Histogram Publishing Method Based on Differential Privacy*, 2018.

[40] M. Hay, V. Rastogi, G. Miklau, and D. Suciu, "Boosting the accuracy of differentially-private queries through consistency," CoRR, 2009, https://arxiv.org/abs/0904.0942.

[41] C. Piao, Y. Shi, J. Yan, C. Zhang, and L. Liu, "Privacy-preserving governmental data publishing: a fog-computing-based differential privacy approach," *Future Generation Computer Systems*, vol. 90, pp. 158–174, 2019.

[42] H. Li, J. Cui, X. Meng, and J. Ma, "IHP: improving the utility in differential private histogram publication," *Distributed and Parallel Databases*, vol. 37, no. 4, pp. 721–750, 2019.

[43] X. Xiao, G. Wang, and J. Gehrke, "Differential privacy via wavelet transforms," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 8, pp. 1200–1214, 2011.

[44] M. Hay, C. Li, G. Miklau, and D. Jensen, "Accurate estimation of the degree distribution of private networks," in *2009 Ninth IEEE International Conference on Data Mining*, pp. 169–178, Miami, FL, USA, December 2009.

[45] J. Xia, W. Huang, Z. Ma, X. Dai, and L. He, "Gradient-based differential privacy optimizer for deep learning model using collaborative training mode," in *2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT)*, Dalian, China, October 2019.

[46] J. Liu, "Security and privacy problems and countermeasures of internet of things applications," in *Proceedings of the 2017 5th International Conference on Mechatronics, Materials, Chemistry and Computer Engineering (ICMMCCE 2017)*, Edmonton, AB, Canada, 2017.

[47] C. Task and C. Clifton, "A guide to differential privacy theory in social network analysis," in *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pp. 411–417, Istanbul, Turkey, August 2012.

[48] R. Chen, B. C. M. Fung, P. S. Yu, and B. C. Desai, "Correlated network data publication via differential privacy," *The VLDB Journal*, vol. 23, no. 4, pp. 653–676, 2014.

[49] W. Jinqiu, Q. Gang, and K. Pengbin, "Emerging 5G multicarrier chaotic sequence spread spectrum technology for underwater acoustic communication," *Complexity*, vol. 2018, Article ID 3790529, 7 pages, 2018.

[50] C.-H. Chen, F. Song, F.-J. Hwang, and L. Wu, "A probability density function generator based on neural networks," *Physica A: Statistical Mechanics and its Applications*, vol. 541, article 123344, 2020.

[51] J. Zhang, Z. Zhang, X. Xiao et al., "Functional mechanism: regression analysis under differential privacy," in *Proceedings of the 38th Conference of Very Large Database*, pp. 1364–1375, Istanbul, Turkey, 2012.

[52] J. Cao, Q. Xiao, G. Ghinita, N. Li, E. Bertino, and K. L. Tan, "Efficient and accurate strategies for differentially-private sliding window queries," in *Proceedings of the 16th International Conference on Extending Database Technology*, pp. 191–202, New York,USA, 2013.

[53] W. Qardaji, W. Yang, and N. Li, "Differentially private grids for geospatial data," in *2013 IEEE 29th International Conference on Data Engineering (ICDE)*, pp. 32-33, Brisbane, Australia, April 2013.

[54] X. J. Zhang, X. J. Zhang, C. C. Fu, and X. F. Meng, "Facial image publication with differential privacy," *Journal of Image and Graphics*, vol. 23, no. 9, pp. 1305–1315, 2018.

[55] T. Dalenius, "Towards a methodology for statistical disclosure control," *StatistikTidskrift*, vol. 15, article 429–222, 1977.

[56] G. Kellaris and S. Papadopoulos, "Practical differential privacy via grouping and smoothing," *Proceedings of the Very Large Database Endowment*, vol. 6, no. 5, pp. 301–312, 2013.

[57] X. Zhang, R. Chen, J. Xu, X. Meng, and Y. Xie, "Towards accurate histogram publication under differential privacy," in *Proceedings of the 2014 SIAM International Conference on Data Mining*, pp. 587–595, Philadelphia, PA, April 2014.

[58] X. Li, J. Yang, Z. Sun, and J. Zhang, "Differentially private release of the distribution of clustering coefficients across communities," *Security and Communication Networks*, vol. 2019, Article ID 2518714, 9 pages, 2019.