IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Traitement du Signal
Vol. 38, No. 5, October, 2021, pp. 1385-1401
Journal homepage: http://iieta.org/journals/ts

Differential Privacy Protection of Face Images Based on Region Growing

Chao Liu[1,2], Jing Yang[1*], Weinan Zhao[2], Yining Zhang[3], Cuiping Shi[2], Fengjuan Miao[2], Jinsong Zhang[2]

[1] College of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China
[2] College of Communication and Electronic Engineering, Qiaihar University, Qiqihar 161000, China
[3] Department of Computer Engineering, DaQing Vocational College, Daqing 163000, China

Corresponding Author Email: yangjing@hrbeu.edu.cn

https://doi.org/10.18280/ts.380514

Received: 3 June 2021
Accepted: 10 September 2021

Keywords:
face image publication, interactive framework, differential privacy, region growing, growth rule

ABSTRACT

Face images, as an information carrier, are rich in sensitive information. Direct publication of these images would cause privacy leak, due to their natural weak privacy. Most of the existing privacy protection methods for face images adopt data publication under a non-interactive framework. However, the ε-effect under this framework covers the entire image, such that the noise influence is uniform across the image. To solve the problem, this paper proposes region growing publication (RGP), an algorithm for the interactive publication of face images under differential privacy. This innovative algorithm combines the region growing technique with differential privacy technique. The privacy budget ε is dynamically allocated, and the Laplace noise is added, according to the similarity between adjacent sub-images. To measure this similarity more effectively, the fusion similarity measurement mechanism (FSMM) was designed, which better adapts to the intrinsic attributes of images. Different from traditional region growing rules, the FSMM fully considers various attributes of images, including brightness, contrast, structure, color, texture, and spatial distribution. To further enhance algorithm feasibility, RGP was extended to atypical region growing publication (ARGP). While RGP limits the region growing direction between adjacent sub-images, ARGP searches for the qualified sub-images across the image, with the aid of the exponential mechanism, thereby expanding the region merging scope of the seed point. The results show that our algorithm can satisfy ε-differential privacy, and the denoised image still have a high availability.

1. INTRODUCTION

Since its nascency, facial recognition has attracted much attention for the intuitiveness, uniqueness, and privacy of face images. The public cautiously enjoy the convenience of facial recognition, while worrying about whether their privacy and rights will be violated someday. With the commercialization of facial recognition, consumers are calling for effective regulation of the technology. Many scholars and organizations have worked to promote relevant legislations. Against this backdrop, the United States Government Accountability Office (U.S. GAO) released two reports: *Facial Recognition Technology: Federal Law Enforcement Agencies Should Better Assess Privacy and Other Risks*, and *Facial Recognition Technology: Privacy and Accuracy Issues Related to Commercial Uses*. In China, the *Public Research Report of Facial Recognition Applications* shows that, more than 60% of the respondents feel that facial recognition has been misused, and over 30% expressed that they have suffered privacy leak or financial loss due to the misuse of facial recognition. Therefore, information technology is a mixed blessing. How to make good use of the technology is a worldwide concern. Owing to the rapid development of information technology and multimedia technology, it is increasingly easy to acquire and share digital face images. Users can upload the photos in their mobile phones or digital cameras to social network platforms like Twitter, Pinterest, LinkedIn, or WeChat. Statistics have shown that more than 3.2 billion face images are shared by users each day on the major social network platforms across the globe. Besides, countless face images are generated from videos. These digital images contain lots of sensitive personal information. If they are collected and analyzed by malicious third-parties, privacy leak and other losses will be beyond measure [1].

In related reports, some stakeholders express concerns about the privacy issues related to the commercial use of facial recognition. The technology may publicly identify and track people without being noticed by them, and may collect, use, and share their personal data. However, some working in the industry argue that facial recognition will not bring new or special privacy risks, and the risks of the technology are controllable.

The National Telecommunications and Information Administration (NTIA) once organized a test on multiple parties. Some participants worried that facial recognition may threaten personal privacy, because it reduces the anonymity of individuals in public or commercial locations (such as pavement or stores). The participants also expressed the concern that facial recognition may be used to individuals in public, and erode personal privacy. Industry insiders revealed that facial recognition has not been widely used to track consumers in business environments.

The Center for Democracy and Technology announces that most people are willing to be facially identified in public by some people or enterprises. But they do not want their faces to be linked with their names, not to mention associating their

faces with other data, such as network behavior, and travel plan. If facial technology is widely adopted in public, and if the recognized faces are shared between enterprises, the consumer movement from one place to another will be tracked any time by a network of cameras. A representative of World Privacy Forum wore that, most consumers will find their privacy be violated, if their activities are tracked by security cameras for marketing. Similarly, the staff of the Federal Trade Commission reported that, consumers will face higher privacy risks, if enterprises start tracking consumers in stores, using collected digital images.

Another widely concerned issue is that consumers will find it impossible to opt out if facial technology is popularized, even if they are notified of the right to reject facial recognition. In many cases, facial recognition is applied to commercial identity recognition or verification, without the knowledge or consent from consumers. This issue has been noticed by some industry trade organizations, which remain cautious about using facial recognition without informed consent from consumers. Computer and Communications Industry Association suggested that, when an enterprise matches face images with names and curriculum vitae through facial recognition, the whole process should remain transparent, and the relevant parties must be given the right to opt out.

Market surveys, patent data, as well as the growing participation in the Facial Recognition Vendor Test (FRVT) organized by the National Institute of Standards and Technology (NIST), indicate that a rising number and types of enterprises are adopting facial recognition. Facial recognition often relies on the acquisition of massive face images (face image databases). Focusing on the massive personal data in such databases, privacy protection organizations, government agencies, the academia, and some industry representatives have expressed concerns over the collection, use, and sharing of personal data by commercial entities.

Face image databases can be sold or shared by all parties, but the selling range remains unknown. The data associated with facial recognition may be sold or shared, often without the knowledge or consent from the affected parties. In recent years, there is a surge in the total amount of personal data collected and shared by resellers and other enterprises, which exacerbates the public concern over this issue.

Privacy is a word with emotional color. Different people have different understandings of this word. According to the International Organization for Standardization (ISO), privacy refers to the features that distinguish individuals or groups from other individuals and groups. Different countries have different legal definitions of privacy. The definition of privacy also varies with objects (individuals, enterprises, governments, etc.). For digital images, sensitive information can be a specific person or object in the original image, a face or fingerprint, embedded information (the location and creation time of the image), or the region of interest (ROI) in the eyes of the image owner. How to publish and analyze images without disclosing sensitive information is the main purpose of privacy protection.

The above privacy issues can be solved through image data query under privacy protection. For the image data released on the social network, the privacy can be usually protected by k-anonymity, access control, and privacy encryption. Fung et al. [2] and Xiao and Tao [3] proposed the k-same method under the anonymization mechanism. Their method anonymizes the published grayscale image, reducing the probability that the attacker re-identifies the user with the published image to less than 1/k. However, the traditional anonymization mechanism has a prominent defect: Before data processing, researchers must set lots of prior conditions for the attacker's background knowledge and attack models. But these conditions are not fully established in reality. For example, Li et al. [4] relied on access control to restrict user access to social network images, and to control the number of transfers and accesses of the images published on the social network. Despite realizing the goal of privacy protection, the access-oriented protection is merely a superficial privacy protection approach for images. If the attacker has a certain background knowledge, he/she will be able to acquire user images and the relevant privacy information. Terrovitis et al. [5] carried out same-state encryption of the grayscale image to prevent the re-identification of the communication process. Nevertheless, the data encryption algorithms are developed based on some assumptions of the attacks. The existing algorithms are quickly phased out, due to the continuously updating attack modes. Sweeney [6] revealed that attackers can identify sensitive personal information (diseases, and address) from anonymous images on Facebook by deriving the social security number (SSN) of the people in the anonymous images from the extra Friendster feature of Facebook. So, is there any technique that can protect personal privacy in spatial data without the background knowledge of an attacker? This is the focus of image data publication based on differential privacy.

In 2006, Dwork [7] invented differential privacy, which disturbs sensitive data by adding noise to the output. Differential privacy can limit the further reasoning ability of the attacker by hiding the impact of a single record, i.e., the output probability of the same result does not change significantly, whether the record is in the dataset. Therefore, differential privacy does not make any assumption for any potential attacker's background knowledge. This is better than other privacy protection technologies. Dwork further investigated differential privacy in a series of theses [8-12], and proposed its implementation mechanism [13, 14]. McSherry [15] pointed out that the differential privacy algorithm for complex privacy issues need to satisfy two combined features: sequence combination and concurrent combination. In recent years, differential privacy is mainly applied in data publication. To protect data publication with differential privacy, the main issues is to ensure the accuracy of the released data or query results, while fulfilling the conditions of differential privacy. The relevant research mainly focuses on the publication mechanism and algorithm complexity. The main research approach is quantification based on calculation theory and learning theory. Based on the realization environment, data publication protected by differential privacy can be divided into interactive data publication and non-interactive data publication [13]. The representative interactive data publication (query) methods are as follows: Roth and Roughgarden [16] presented a Median algorithm that can respond to multiple queries. Hardt and Rothblum [17] developed a private multiplicative weights (PMW) mechanism that increases the number of queries. Gupta et al. [18] put forward a universal iterative dataset creation (IDC) framework. Fan and Xiong [19] designed a FAST algorithm based on filtering and adaptive sampling. Kellaris et al. [20] designed a flow data publication algorithm with an unlimited number of queries. When it comes to non-interactive data publication, histogram publication is the most widely adopted technique. The representative strategies include Xiao et al.'s [21] Privelet algorithm, Xu et al.'s [22]

Noise First and Structure First algorithms, Li et al.'s [23] matrix mechanism, and Yuan et al.'s data- and workload-aware (DAWA) algorithm [24]. Due to the complexity of image data, the research of sensitive information in the images protected by differential privacy is still in the exploratory stage.

The real domain matrix is a common representation of images. Any pixel in the image can be mapped to a value at the corresponding position of the two-dimensional (2D) matrix. The most direct approach is to add a Laplace noise to all values in the matrix. Although this approach can satisfy ε-differential privacy, the disturbed image will be excessively distorted and weakly available. Fourier transform and wavelet transform are often adopted to compress images. Zhang et al. [25] proposed an image compression method based on discrete Fourier transform, which adds corresponding Laplace noise to the compressed image. Despite reducing the noise error, their method introduces reconstruction error to image compression. Considering the noncorrelation between data in image matrix, Nissim et al. [26] converted the grayscale matrix into a one-dimensional (1D) sequential data flow through image segmentation, modeled the data flow with sliding window model, and dynamically distribute privacy budget based on the data similarity between adjacent sliding windows. In this way, the privacy protection of images is achieved. However, this highly feasible approach faces two problems: the overall operation is confined in the 1D space, and Laplace noise covers the entire image.

2. BACKGROUND

2.1 Differential privacy

Dalenius pointed out a problem with the statistics database: No one should get any information about any person by accessing the database. However, absolute privacy protection is impossible due to background knowledge. Differential privacy circumvents this problem, and pursues relative privacy protection, trying to limit any possible privacy leak to the range of a small multiplier factor. Note that severe leaks may still occur, but will not be caused by whether a certain data exists in the database.

Face images, as a carrier of information, are usually stored and transmitted using three-dimensional (3D) matrices (i.e., a color image can be described as three 2D matrices red R, greed G, and blue B). For simplicity, the 3D image matrix is normalized to obtain the corresponding 2D grayscale matrix. Then, image X can be represented as a 2D matrix X_{mn}, where m and n are the number of rows and columns of the matrix, respectively:

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

Adjacent datasets are an important concept of differential privacy. In fact, the concept comes from the basic operations of sets. The main set operation adopted for adjacent datasets is the operation of symmetric difference $\oplus$. In the set operation formula $T = R \oplus S = (R \cup S) - (R \cap S)$, T contains the elements in set R but not in set S, and the elements in set S but not in set R. The number of different elements between the two sets can be calculated by $\Delta = |R \oplus S|$. Before giving a formal definition of differential privacy, it is necessary to define the adjacent datasets of face image X.

Definition 1. Adjacent datasets of face image

For a given image X, the grayscale matrix X_{mn} can be obtained by normalizing the image. Then, there exists $X|X_{mn} = \begin{bmatrix} x_{11}, x_{12}, & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1}, x_{m1}, & \cdots & x_{mn} \end{bmatrix}$, where x_{ij} in matrix X_{mn} represents the grayscale of the corresponding element. If there exists an X' with only one element difference from X, $|X - X'| = x_{ij}(1 \le i \le m, 1 \le j \le n)$, then X and X' are adjacent datasets.

Definition 2. Differential privacy

For a given random algorithm M of image data publication, with the output range of Range(M), the algorithm can provide ε- differential privacy, if its arbitrary outputs on two adjacent grayscale images X and X' satisfy:

$$Pr[M(X) \in S] \le exp(\varepsilon) \times Pr[M(X') \in S] \quad (2)$$

where, ε is typically a small positive number that balances privacy with accuracy. If ε is small, the privacy is high and accuracy is low. The inverse is also true. Normally, ε is selected by the user by executing a privacy policy. When the adjacent datasets vary by only one record, the algorithm satisfies ε-differential privacy. When the adjacent datasets vary by k records, the algorithm satisfies kε-differential privacy.

To realize differential privacy, a certain amount of random noise needs to be added to the query results. Intuitively, the magnitude of the additive noise should surpass 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. Global sensitivity

Let Q be a random query function meeting $Q: D \to R^n$. Then, the global sensitivity of Q can be expressed as:

$$\Delta Q_{GS} = \max_{X,X'} ||Q(D) - Q(D')||_\rho \quad (3)$$

Global sensitivity can indeed be applied to all methods of differential privacy. But some functions have a relatively high global sensitivity. In this case, lots of noise need to be added to ensure privacy. Then, the balance between information availability and privacy will be undermined after privacy protection. To solve the problem, Nissim et al. proposed the concept of local sensitivity [26]. Nissim held that, for the same function, it is possible to predict the distribution feature of a subset d of a dataset D with a high global sensitivity. Under the effect of the prediction result, a relatively small local sensitivity could be obtained specifically for d. The local sensitivity, and its relationship with global sensitivity are explained below.

Definition 4. Local sensitivity

Let Q be a random query function meeting $Q: d \to R^n (d \in D)$. Then, the local sensitivity of Q can be expressed as:

$$\Delta Q_{LS} = \max_{X,X'} ||Q(d) - Q(d')||_\rho \quad (4)$$

$$\Delta Q_{GS} = max\ (\Delta Q_{LS}) \quad (5)$$

Note that any dataset d with predictable distribution feature faces the risk of privacy leak. To solve the potential privacy leak of local sensitivity, the smooth upper bound is defined for local sensitivity.

Definition 5. Smooth upper bound

For a dataset d and its adjacent dataset d', if the local sensitivity of function Q is ΔQ_{LS}, the function $S: d \to R^n$ satisfying $S(d) \geq \Delta Q_{LS}$ and $S(d) \leq e^{\beta} S(d')(\beta > 0)$ is the β-smooth upper bound of the local sensitivity of function Q.

Any function S meeting Definition 5 can be treated as a smooth upper bound. During the use, the local sensitivity ΔQ_{LS} is substituted to function S to obtain the corresponding smoothness sensitivity, and then derive the final Laplace noise. Figure 1 shows the relationship between the smooth upper bound and local sensitivity.

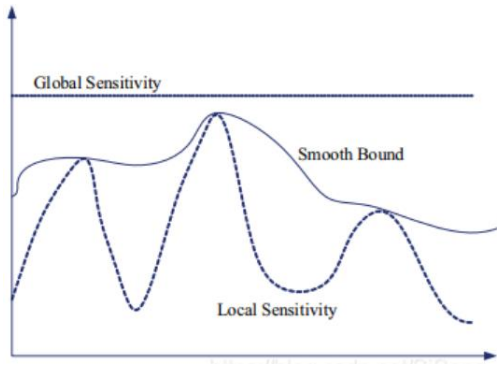

Figure 1. Smooth upper bound of local sensitivity

Theorem 1. Laplace mechanism

Let Q be a query series of the length d. The random algorithm M receives database D and outputs the following vector that satisfy ε-differential privacy:

$$M(D) = Q(D) + < Lap_1(\Delta Q_{LS}/\varepsilon, \ldots, Lap_d(\Delta Q_{LS}/\varepsilon > \quad (6)$$

As the most common noise addition mechanism, the Laplace mechanism disturbs the real output by adding the noise generated by Laplace distribution, thereby achieving differential privacy. The probability density function (PDF) of its noise distribution satisfies $f(x|\mu, b) = \frac{1}{2b} e^{\frac{|x-\mu|}{b}}$ (Figure 2).

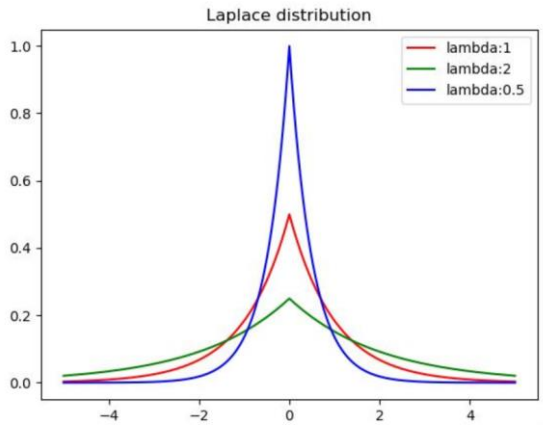

Figure 2. Laplace PDF

Then, the cumulative probability distribution function can be derived from the PDF in the following manner:

$$F(x|\mu, b) = \begin{cases} \frac{1}{2} e^{\frac{\mu - x}{b}}, x < u \\ 1 - \frac{1}{2} e^{\frac{x-\mu}{b}}, x \geq u \end{cases}$$

If $x < \mu$, and $f(x|\mu, b) = \frac{1}{2b} e^{\frac{|x-\mu|}{b}}$, there exists:

$$F(x|\mu, b) = \frac{1}{2b} \int_{-\infty}^{x} e^{\frac{\mu - x}{b}} dx = \frac{1}{2b} \int_{-\infty}^{x} e^{\frac{x-\mu}{b}} dx$$

Suppose $t = \frac{x-\mu}{b}$. We have:

$$F(x|\mu, b) = \frac{1}{2b} \int_{-\infty}^{\frac{x-\mu}{b}} b e^t dt = \frac{1}{2} \int_{-\infty}^{\frac{x-\mu}{b}} e^t dt = \frac{1}{2} [e^t]_{-\infty}^{\frac{x-\mu}{b}} = \frac{1}{2} e^{\frac{x-\mu}{b}}$$

Expectation and variance are the two most basic attributes of the probability distribution. For a Laplace distribution function, the expectation and variance μ and $2b^2$ can be derived by:

Expectation:

$$\begin{aligned} E(x) &= \frac{1}{2b} \left(\int_{-\infty}^{\mu} x e^{\frac{\mu - x}{b}} dx + \int_{\mu}^{+\infty} x e^{\frac{\mu - x}{b}} dx \right) \\ &= \frac{1}{2b} \left(\int_{-\infty}^{0} b(bt + \mu) e^t dt \right. \\ &\quad \left. - \int_{0}^{-\infty} b(\mu - bt) e^t dt \right) \\ &= \frac{1}{2b} \int_{0}^{-\infty} b\big((bt + \mu) + (\mu - bt)\big) e^t dt \\ &= \int_{-\infty}^{0} \mu e^t dt = \mu \end{aligned}$$

Variance:

$$\begin{aligned} D(x) &= E(x^2) - E^2(x) \\ &= \frac{1}{2b} \left(\int_{-\infty}^{\mu} x^2 e^{\frac{\mu - x}{b}} dx + \int_{\mu}^{+\infty} x^2 e^{\frac{\mu - x}{b}} dx \right) \\ &\quad - \mu^2 \\ &= \frac{1}{2b} \left(\int_{-\infty}^{0} b((bt + \mu)^2 \right. \\ &\quad \left. + (\mu - bt)^2) e^t dt \right) - \mu^2 \\ &= \frac{1}{2b} \left(\int_{-\infty}^{0} 2b(b^2 t^2 + \mu^2) e^t dt \right) - \mu^2 \\ &= b^2 \int_{-\infty}^{0} t^2 e^t dt = b^2 \int_{-\infty}^{0} t^2 de^t \\ &= b^2 \left([t^2 e^t]_{-\infty}^{0} - \int_{-\infty}^{0} e^t dt^2 \right) \\ &= -2b^2 \int_{-\infty}^{0} t de^t \\ &= -2b^2 \left([te^t]_{-\infty}^{0} - \int_{-\infty}^{0} e^t dt \right) = 2b^2 \end{aligned}$$

Note that $Lap_i(\Delta Q_{LS}/\varepsilon)$ ($1 \leq i \leq d$) is an independent Laplace noise, whose magnitude is positively proportional to ΔQ_{LS}, and negatively proportional to ε.

Theorem 2. Exponential mechanism

The exponential mechanism mainly handles the non-numeric outputs of the sampling algorithm. Under any exponential mechanism, the sampling algorithm M satisfies ε-differential privacy if it meets:

$$M(X,u)=\left\{r:\left|Pr[r\in S]\propto exp\left(\frac{\varepsilon u(X,r)}{2\Delta u_{LS}}\right)\right.\right\} \quad (7)$$

where, $u(X,r)$ is the scoring function; Δu_{LS} is the global sensitivity of the scoring function $u(X,r)$; S is the output domain of our algorithm; r is the selected output term of the output domain S. The higher the score of $u(X,r)$, the greater the probability for r being selected as the output.

Corollary 1. Algorithm M in Definition 2 satisfies ε-differential privacy.

Proof: X and X′ are adjacent datasets, with $M(X)=(x_1,x_2,\dots x_d)^T$ and $M(X')=(x'_1,x'_2,\dots x'_d)^T=(x_1+\Delta x_1,x_2+\Delta x_2,\dots x_d+\Delta x_d)^T$. Without loss of generality, it is assumed that $x_i=0(x_i\in X)$ and $\rho=1$. It can be calculated that $M(X)=(0,0,\dots 0)^T$ and $M(X')=(\Delta x_1,\Delta x_2,\dots \Delta x_d)^T$. When the vector $S=(s_1,s_2,\dots s_d)^T$ is outputted, it can be seen that:

$$Pr[M(X)\in S]=\prod_{i=1}^{d}\frac{\varepsilon}{2\Delta Q_{GS}}e^{\frac{\varepsilon}{\Delta Q_{GS}}|s_i|}$$

$$Pr[M(X')\in S]=\prod_{i=1}^{d}\frac{\varepsilon}{2\Delta Q_{GS}}e^{\frac{\varepsilon}{\Delta Q_{GS}}|\Delta x_i-s_i|}$$

$$\frac{\Pr[M(X)\in S]}{Pr[M(X')\in S]}=\frac{\prod_{i=1}^{d}\frac{\varepsilon}{2\Delta Q_{GS}}e^{\frac{\varepsilon}{\Delta Q_{GS}}|s_i|}}{\prod_{i=1}^{d}\frac{\varepsilon}{2\Delta Q_{GS}}e^{\frac{\varepsilon}{\Delta Q_{GS}}|\Delta x_i-s_i|}}=\prod_{i=1}^{d}e^{\frac{\varepsilon}{\Delta Q_{GS}}(|s_i|-|\Delta x_i-s_i|)}=e^{\frac{\varepsilon}{\Delta Q_{GS}}\sum_{i=1}^{d}(|\Delta x_i-s_i|-|s_i|)}$$

Because of inequality $-|\Delta x_i|\le|\Delta x_i-s_i|-|s_i|\le|\Delta x_i|$, and $\Delta Q_{GS}=\max\left(\sum_{i=1}^{d}|x_i-x'_i|\right)=\max\left(\sum_{i=1}^{d}|\Delta x_i|\right)$, we have $\sum_{i=1}^{d}(|\Delta x_i-s_i|-|s_i|)\le\Delta Q_{GS}$. Hence, Algorithm M satisfies ε- differential privacy.

Corollary 2. When norm p takes a random value, Algorithm M does not necessarily satisfy ε- differential privacy.

Proof: From the definition of the norm, it can be learned that $\|x\|_p$ is a decreasing function with the increase of p. Let $\Delta Q_{GS}^{(p)}=\max_{X,X'}\|M(X)-M(X')\|_p$ be the value of ΔQ_{GS} under norm p, with $1\le p\le+\infty$. Suppose $\Delta Q_{GS}^{(p)}$ is a decreasing function, i.e., $\Delta Q_{GS}^{(p)}\le\Delta Q_{GS}^{(1)}$. Then, when the norm is p,

$$M(D)=Q(D)+\left(Lap_1\left(\frac{\Delta Q_{GS}}{\varepsilon}\right),Lap_2\left(\frac{\Delta Q_{GS}}{\varepsilon}\right),\dots Lap_d\left(\frac{\Delta Q_{GS}}{\varepsilon}\right)\right)^T$$

$$M(D)=Q(D)+\left(Lap_1\left(\frac{\Delta Q_{GS}}{\varepsilon\times\frac{\Delta Q_{GS}^{(1)}}{\Delta Q_{GS}^{(p)}}}\right),Lap_2\left(\frac{\Delta Q_{GS}}{\varepsilon\times\frac{\Delta Q_{GS}^{(1)}}{\Delta Q_{GS}^{(p)}}}\right),\dots Lap_d\left(\frac{\Delta Q_{GS}}{\varepsilon\times\frac{\Delta Q_{GS}^{(1)}}{\Delta Q_{GS}^{(p)}}}\right)\right)^T$$

Since $M(D)$ satisfies $\varepsilon\times\frac{\Delta Q_{GS}^{(1)}}{\Delta Q_{GS}^{(p)}}$ differential privacy, with $\varepsilon\times\frac{\Delta Q_{GS}^{(1)}}{\Delta Q_{GS}^{(p)}}\ge\varepsilon$, we have:

$$\frac{Pr[AM(X)\in S]}{Pr[M(X')\in S]}\le e^{\varepsilon}\times\frac{\Delta Q_{GS}^{(1)}}{\Delta Q_{GS}^{(p)}}$$

Therefore, Corollary 2 holds.

There are two combinations of differential privacy mechanism: serial combination and parallel combination. For the protection of differential privacy, simple differential privacy algorithms can be combined in these two approaches to obtain innovative complex differential privacy algorithms.

Property 1. Differential privacy-serial combination property

For a given dataset X and a set of differential privacy algorithms $M_1(X),M_2(X),\dots,M_m(X)$ related to X, if algorithm $M_i(D)$ satisfies ε_i - differential privacy, and the random processes of any two algorithms are independent of each other, then the algorithm combined from these algorithms satisfies $\sum_{i=1}^{m}\varepsilon_i$- differential privacy.

Proof: Since algorithm $M_i(D)$ satisfies ε_i - differential privacy:

$$\forall S_i\in Range(M_i),Pr[M_i(D)=S_i]\le e^{\varepsilon\times|X\oplus X'|}\times Pr[M_i(X')=S_i] \quad (8)$$

Suppose algorithm $\overrightarrow{M}(X)$ is the algorithm combined from all algorithms $M_i(X)$. Let $\{S_1,S_2,\dots,S_m\}$ denote the output of algorithm $\overrightarrow{M}(X)$.

Since the random processes of any two algorithms $M_i(D)$ are independent of each other, the following formula holds:

$$\forall S_i=\{S_1,S_2,\dots,S_m\}Range\left(\overrightarrow{M}\right),Pr\left[\overrightarrow{M}(X)=S\right]=\prod_{i=1}^{m}Pr[M_i(D)=S_i]$$

Hence, the following must be independent:

$$Pr\left[\overrightarrow{M}(X)=S\right]=\prod_{i=1}^{m}Pr[M_i(D)=S_i]\le\prod_{i=1}^{m}(e^{\varepsilon_i}\times Pr[M_i(D')=S_i])=e^{\sum_{i=1}^{m}\varepsilon_i}\times\prod_{i=1}^{m}Pr[M_i(D')=S_i]=e^{\sum_{i=1}^{m}\varepsilon_i}\times Pr\left[\overrightarrow{M}(X)=S\right]$$

From the above, it is possible to derive:

$$\Pr\left[\overrightarrow{M}(X)=S\right]\le e^{\sum_{i=1}^{m}\varepsilon_i}\times\Pr\left[\overrightarrow{M}(X')=S\right] \quad (9)$$

According to the definition of differential privacy, algorithm $\overrightarrow{M}(X)$ satisfies $\sum_{i=1}^{m}\varepsilon_i$- differential privacy.

Property 2. Differential privacy-parallel combination property

Let $M_1(X_1),M_2(X_2),\dots,M_m(X_m)$ be a series of ε - differential privacy algorithms with input datasets $X_1,X_2,\dots,X_m$, respectively. Suppose the random processes of

any two algorithms are independent of each other. Then, the algorithm combined from these algorithms satisfies ε - differential privacy.

Proof: Since all algorithms $M_i(X_i)$ satisfy ε- differential privacy, the complete differential privacy can be defined as:

$$\forall S_i \in Range(M_i), \Pr[M_i(X_i) = S_i] \leq e^{\varepsilon \times |X_i \oplus X_{i'}|} \times \Pr[M_i(X') = S_i]$$

Let $\overrightarrow{M}(X)$ be the algorithm combined from algorithms $M_i(X_i)$, and $\{S_1, S_2, \dots, S_m\}$ be its output. Since the random processes of any two algorithms $M_i(X_i)$ algorithms are independent of each other, the following must be independent:

$$\forall \boldsymbol{S} = \{S_1, S_2, \dots, S_m\} Range(\overrightarrow{M}), Pr[\overrightarrow{M}(X) = S] = \prod_{i=1}^{m} Pr[M_i(X_i) = S_i]$$

From the above, it is possible to derive:

$$\begin{aligned} Pr[\overrightarrow{M}(X) = S] &= \prod_{i=1}^{m} Pr[M_i(X_i) = S_i] \\ &\leq \prod_{i=1}^{m} \left(e^{\varepsilon \times |X_i \oplus X_{i'}|} \times Pr[M_i(X') = S_i]\right) \\ &= e^{\varepsilon \times \sum_{i=1}^{m} |X_i \oplus X_{i'}|} \times \prod_{i=1}^{m} Pr[M_i(X'_i) = S_i] \\ &= e^{\varepsilon \times \sum_{i=1}^{m} |X_i \oplus X_{i'}|} \times Pr[\overrightarrow{M}(X') = S] \end{aligned}$$

From the leftmost or rightmost expression, we have:

$$Pr[\overrightarrow{M}(X) = S] \leq e^{\varepsilon \times \sum_{i=1}^{m} |D_i \oplus D_{i'}|} \times Pr[\overrightarrow{M}(X') = S] \quad (10)$$

The above formula shows that the differential privacy algorithm under parallel combination meets $(\varepsilon \times \sum_{i=1}^{m} |D_i \oplus D_{i'}|)$- differential privacy.

Besides, X and X′ should satisfy $|X \oplus X'| = 1$, according to the precondition of the definition of differential privacy. Since $\forall i \neq j, X_i \cap X_j = \emptyset \wedge X'_i \cap X'_j = \emptyset$, it can be derived for $\sum_{i=1}^{m} |X_i \oplus X_{i'}|$ that:

$$\begin{aligned} \sum_{i=1}^{m} |X_i \oplus X_{i'}| &= \left|\bigcup_{i=1}^{m} X_i \oplus X_{i'}\right| \\ &= \left|\bigcup_{i=1}^{m} \left((X \cap R_i) \oplus (X' \cap R_i)\right)\right| \\ &= \left|\bigcup_{i=1}^{m} \left((X \oplus X') \cap R_i\right)\right| \\ &= \left|(X \oplus X') \cap \bigcup_{i=1}^{m} R_i\right| \\ &= (X \oplus X') \cap R \end{aligned}$$

R is the definition domain of elements, with $D \subseteq R, D' \subseteq R$. The final result of the above derivation is:

$$\sum_{i=1}^{m} |D_i \oplus D_{i'}| = |D \oplus D'| = 1 \quad (11)$$

Therefore, the differential privacy algorithm under parallel combination satisfies ε- differential privacy.

Corollary 3. Suppose $M_1(X_1), M_2(X_2), \dots, M_m(X_m)$ are a series of independent differential privacy algorithms, and algorithm $M_i(D)$ satisfies ε_i-differential privacy. Then, the algorithm combined from these algorithms satisfy $\max_{1>i>m} \varepsilon_i$-differential privacy.

Proof: Formula (11) shows that $\sum_{i=1}^{m} |X_i \oplus X'_i| = |X_i \oplus X'_i| = 1$. In all $|X_i \oplus X'_i|$, there is one and only one $|X_i \oplus X'_i|$ equaling 1, and the other $|X_i \oplus X'_i| = 0 (j \neq i)$. In this case, X_i and X'_i are adjacent datasets, $X_j = X'_{j'}$.

Let $Q = \langle X, X' \rangle$ denote the dataset combined from X and X′. According to the above property, Q can be divided into m classes $Q_1, Q_2, \dots, Q_m$, according to the subpart difference between X and X′. The Q_i in each class can be defined as:

$$Q_i = \{\langle X, X' \rangle | (|X_i \oplus X'_i| = 1) \wedge (|X_j = X'_j|)\}$$

Suppose the combined algorithm $\overrightarrow{M}(X)$ with unknown degree of protection satisfies ε′- differential privacy.

$$\forall X, X', \text{ satisfying } |X_i \oplus X'_i| = 1, \Pr[\overrightarrow{M}(X) \in \boldsymbol{S}] \leq \Pr[\overrightarrow{M}(X') \in \boldsymbol{S}]$$

$$\Leftrightarrow \bigwedge_{1<i<m} \forall X, X', \langle X, X' \rangle \in Q_i, \Pr[\overrightarrow{M}(X) \in \boldsymbol{S}] \leq \Pr[\overrightarrow{M}(X') \in \boldsymbol{S}]$$

It is known that each constituent algorithm $M_i(X)$ satisfies ε_i-differential privacy. By definition, it can be derived that $\Pr[\overrightarrow{M}(X) \in S] \leq e^{\varepsilon_i} \times \Pr[\overrightarrow{M}(X') \in S]$ holds if and only if $\varepsilon' \geq \varepsilon_i$. Thus, ε′ can be calculated by:

$$\varepsilon' = min\{\varepsilon' | \wedge_{1<i<m} (\varepsilon' \geq \varepsilon_i)\} = min\left\{\varepsilon' \middle| \varepsilon' \geq \max_{1<i<m} \varepsilon_i\right\} = \max_{1<i<m} \varepsilon_i$$

Therefore, $\overrightarrow{M}(X)$ satisfies $\max_{1>i>m} \varepsilon_i$-differential privacy.

2.2 Region growing

Region growing is an ancient method of image segmentation. The earliest region growing segmentation was proposed by Adams and Bischof [27]. There are two ways to implement the region growing segmentation. The first way is to select a small block or seed point from the target object, add the surrounding pixels to the seed point continuously by a certain rule, and eventually combine all the pixels representing the object into one region. The second way is to segment the image into multiple highly consistent small blocks (e.g., small blocks with the same grayscale), and merge the blocks into large blocks by a certain rule, thereby achieving the goal of image segmentation. One of the typical region growing methods is Matalas et al.'s [28] facet model-based regional growing strategy. Over segmentation, i.e., segmenting the image into too many regions, poses an intrinsic defect of region growing.

Region growing can segment the original image into a series of regions. This simple method can segment connected regions with the same features, and clearly delineate the edges between the regions. Region growing is capable of achieving the optimal performance, in the absence of prior knowledge. Therefore, the method applies to the segmentation of complex images, i.e., natural scenery images and face images.

The basic idea of region growing is to combine pixels with similar properties into a region. Region growing can be realized in the following steps:

Step 1. Choose a region or pixel in the original image as the seed point, which serves as the starting point of growth. The seed point can be selected randomly or based on specific demands. Normally, the seed point should not cover any sudden change of pixels.

Step 2. Compare the seed point with each surrounding region that has not been merged. If the two have the same or similar attributes, merge them into one seed point (the necessity of merging depends on the preset growth rule or similarity rule). Then, treat the merged seed point as the new seed point, and repeat the above process, until no qualified region can be merged. In this way, a region is fully grown. The above operation segments the original image into a labeled region and an unlabeled region.

Step 3. Choose a region in the unlabeled region as a new seed point, and repeat Step 2.

Step 4. Repeat Step 3 until the entire image no longer contains any non-seed region to be merged. This marks the completion of region growth of the entire image.

To ensure the accuracy of initial segmentation, the regions should not be too large. The region size needs to be limited in the growth rule. That is, a threshold num should be defined for the region size. If a region grows larger than num, the region must stop growing. After region growing, some small regions (e.g., those with fewer than 10 pixels) will be merged with the most similar adjacent region, aiming to reduce the number of vertices in the subsequent graph.

3. METHODOLOGY

Today's differential privacy methods for face image publication mostly change the data in the original matrix $X_{m\times n}$ through reconstruction (e.g., Fourier transform or wavelet transform), disturb the changed data by adding Laplace noise, and restore the disturbed data to obtain a noisy matrix $X_{m\times n}'$. However, two errors may occur during the derivation of $X_{m\times n}'$: the noise error $LE(X_{m\times n}')$ brought by the Laplace mechanism, and the reconstruction error $RE(X_{m\times n}')$ produced in the reconstruction of the original data. Hence, the overall error $Error(X_{m\times n}')$ of the released face image $X_{m\times n}'$ can be expressed as:

$$Error(X_{m\times n}')= LE(X_{m\times n}')+ RE(X_{m\times n}') \tag{12}$$

To reduce noise, the above-mentioned reconstruction methods (Fourier transform and wavelet transform) essentially modify the 2D data extracted from the face image, and add noise to the modified data. None of these methods can avoid the reconstruction error $RE(X_{m\times n}')$.

Therefore, this paper inherits the approach of Liu et al. [1]: rather than change the data in the 2D matrix, implement the reconstruction from the perspective of structure. Without changing the original data, this approach effectively avoids $RE(X_{m\times n}')$. Hence, the following conclusion can be drawn:

$$Error(X_{m\times n}')= LE(X_{m\times n}') \tag{13}$$

On differential privacy of data publication, the main research focus lies in the publication method of non-interactive data. In this method, the ε effect range covers the entire image, such that the noise level is the same across the image. In reality, however, the sensitive information of a face image clusters in specific regions. For an image, different regions need different degrees of privacy protection. Therefore, this paper tries to protect the privacy of face images by combining the differential privacy of interactive data publication with the image segmentation technology. Under the premise of meeting ε-differential privacy, the integrated method reduces the influence of Laplace noise on the privacy protection image, and strikes a balance between image availability and degree of privacy protection.

3.1 LAP algorithm

This paper designs the LAP algorithm based on Laplace mechanism. Without changing the original data, the LAP algorithm directly disturbs the values in the 2D matrix of the original image with Laplace noise, and publishes the disturbed image straightforwardly.

Drawing on image segmentation theory, each pixel x_{ij} in the grayscale matrix $X_{m\times n}$ of the original image is treated as an independent entity. The pixels do not interfere with each other, and the privacy budget is distributed evenly. Then, each $x_{ij}(1\le i\le m,\ 1\le j\le n)$ consumes the privacy budge of the size $\varepsilon/(m\times n)$. In the LAP algorithm, the overall error induced by Laplace noise can be expressed as:

$$\begin{aligned} Error(X_{m\times n}') &= E\sum_{i=1}^{i=m}\left(\sum_{j=1}^{j=n}\left(x_{ij}'-x_{ij}\right)^2\right) \\ &= E\sum_{i=1}^{i=m}\left(\sum_{j=1}^{j=n}\left(x_{ij}-x_{ij}\right.\right. \\ &\quad \left.\left.+ lap\left(\Delta Q\times m\times\frac{n}{\varepsilon}\right)\right)^2\right) \\ &= 2mn(\Delta Q\times m\times n/\varepsilon)^2 \end{aligned} \tag{14}$$

Although the LAP algorithm satisfies ε-differential privacy, a huge noise error will occur when the algorithm is adopted to protect image privacy, if the image is too large. In this case, the noisy image will be weakly available.

3.2 Publication method based on region growing

To control the effect of noise error on the privacy protection of image publication, this paper proposes an image publication algorithm called region growing publication (RGP), which combines region growing with differential privacy technique. In traditional region growing rule, the mean intensity difference between the seed point and a surrounding pixel is compared with the given threshold to judge if the pixel should be merged into the seed point. That is, the growth rule can be expressed as:

$$|x_{i\pm1,j\pm1}-x_{ij}^R|<Th \tag{15}$$

where, $x_{i\pm1,j\pm1}$ is the grayscale of the surrounding pixel; x_{ij}^R is the mean intensity of region R with x_{ij} as the seed point; Th is the given threshold. Formula (15) reveals two deficiencies with the traditional region growing method. Firstly, the object to be merged is merely a pixel, which carries very limited information, as the most basic unit of the image. The lack of information will bias the final result of region merging. To

solve the problem, this paper divides the face image $X_{m\times n}$ into multiple non-intersecting sub-images:

$$X_{m\times n}=\begin{bmatrix} x_{11}, & x_{12}, & \cdots & x_{1n} \\ \vdots & & \ddots & \vdots \\ x_{m1}, & x_{m1}, & \cdots & x_{mn} \end{bmatrix} = \begin{bmatrix} T_{11}, & T_{12}, & \cdots & T_{1J} \\ \vdots & & \ddots & \vdots \\ T_{I1}, & T_{I1}, & \cdots & T_{IJ} \end{bmatrix} \tag{16}$$

Any sub-image $T_{ij}(1 \le i \le I, 1 \le j \le J)$ contains multiple pixels. Hence, the non-intersecting sub-images T_{ij} could carry lots of information of the original image. The division is nondestructive and reversible (Figure 3).

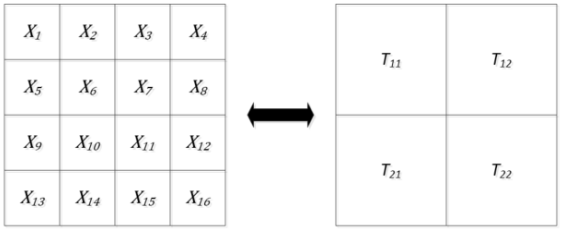

Figure 3. Dividing the original image into multiple sub-images

Secondly, the key of region growing is the definition of growth rule. Due to the natural complexity of face images, the growth rule should be configured in the light of the various attributes of the original image. The traditional growth rule mainly relies on the grayscale difference. However, the grayscale of a single pixel cannot reflect the rich information of the original image. To solve the problem, this paper takes sub-images as the basic unit of region growing. The sub-images retain most of the features of the original image, namely, brightness, contrast, structure, color, texture, and spatial distribution. To further improve the accuracy of region merging with sub-images as the basic unit, this paper puts forward a brand-new growth rule called fusion similarity measurement mechanism (FSMM).

After being converted into a grayscale matrix, the original image becomes meaningful in mathematical calculation. Let X and Y be the grayscale matrices of two adjacent sub-images, respectively (During region growing, sub-image Y can be any of the eight adjacent sub-images of sub-image X), $X=\begin{bmatrix} X_1 & \cdots & \cdots \\ \cdots & \ddots & \cdots \\ \cdots & \cdots & X_c \end{bmatrix}$, $Y=\begin{bmatrix} Y_1 & \cdots & \cdots \\ \cdots & \ddots & \cdots \\ \cdots & \cdots & Y_c \end{bmatrix}$. Then, the similarity between X and Y can be measured by Euclidean distance:

$$d(X,Y)=\sqrt[2]{\sum_{i=1}^{n}(X_i-Y_i)^2} \tag{17}$$

Many scholars have adopted Euclidean distance to measure the similarity between two matrices, and to regulate region growing. The measurement can accurately characterize the difference between two sub-images in mathematical properties. This difference is directly exhibited in terms of image colors. However, the similarity rule, which solely considers the mathematical difference between grayscale matrices, cannot fully demonstrate the other properties of the image.

Below is an example illustrating the limitation of Euclidean distance as region growing rule: Suppose there are three grayscale matrices $T_1=\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$, $T_2=\begin{bmatrix} 7 & 6 & 3 \\ 15 & 20 & 11 \\ 6 & 3 & 5 \end{bmatrix}$, and $T_3=\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 255 \end{bmatrix}$. The Euclidean distances between T_1 and T_2, and T_1 and T_3 are denoted as θ_1 and θ_2, respectively. Since $\theta_1 < \theta_2$, T_1 is more similar to T_2. However, after the matrices are converted to grayscale images, T_1 is more similar to T_3 even in naked eyes. In this example, images T_1 and T_3 have very similar colors. However, their Euclidean distance (color eigenvalue) is exaggerated by a salient eigenvalue. The mis-judgement is attributed to the fact that Euclidean distance (color eigenvalue) is solely considered to compute the grayscale matrices of the two images. The salient value of T_3 could be the result of texture or shape features of the image.

To solve the above problem, the Jaccard similarity index is introduced to compute the similarity θ between the grayscale matrices of the two images. The Jaccard similarity index, a metric of the similarity between two sets, is defined as the number of pixels in the interaction between two sets, divided by the number of pixels in the union of two sets:

$$\rho(X,Y)=|X\cap Y|/|X\cup Y| \tag{18}$$

In this paper, the Jaccard similarity index is taken as the optimization value in the result of formula (17):

$$\theta=d(X,Y)/\rho(X,Y)+\sigma \tag{19}$$

where, $d(X,Y)$ is the Euclidean distance between the grayscale matrices of two images; $\rho(X,Y)$ is the Jaccard similarity index between the grayscale matrices of two images; σ is a small positive number that prevents the denominator from being zero.

The above-mentioned example can prove that the θ value derived from both Euclidean distance and Jaccard similarity index can largely reflect the similarity between the grayscale matrices of two images, and serve as the growth rule. However, some special cases are found during the experiments, in which the above formula is not applicable. Another example was given to explain the influence of the special cases.

Suppose two local grayscale matrices $T_1=\begin{bmatrix} 7 & 9 & 8 \\ 4 & 3 & 7 \\ 1 & 5 & 6 \end{bmatrix}$, and $T_2=\begin{bmatrix} 8 & 7 & 9 \\ 3 & 4 & 7 \\ 6 & 5 & 1 \end{bmatrix}$ are obtained from the same image. The similarity θ between them, which is calculated by formula (19), is far smaller than the expectation θ′ given during the explements. θ′ is a preset threshold. If $\theta > \theta'$, the grayscale matrices are not similar; otherwise, the grayscale matrices are similar. It can be observed from the experimental values in matrices T_1 and T_2 that formula (19) cannot calculate the correct θ value, due to its neglection of spatial distribution features of the images. Hamming distance is often employed to measure the difference between two spatial vectors of the same structure and size. This criterion sequentially compares the values in the corresponding positions of two spatial vectors X and Y. If $X_i=Y_i$, value is 0; otherwise, the value is 0. After all the corresponding positions of two spatial vectors have been compared, all the ones will be cumulated to obtain the Hamming distance between the two spatial vectors:

$$\mu(X,Y) = \sum_{i=1}^{n} X_i \oplus Y_i \tag{20}$$

In this paper, the values are extracted from each grayscale matrix by formula (1). Thus, the values in the grayscale matrices are approximate. The color information of the images cannot be truthfully reflected, with Hamming distance as the only criterion. What is worse, the Hamming distance cannot be directly applied to formula (19).

To solve these problems, Hamming distance is combined with the result of perceptual hash algorithm into a disturbance φ for formula (19), further improving the correctness of the similarity θ between grayscale matrices. Combining formulas (19) and (20), the similarity θ between the grayscale matrices of two images can be expressed as:

$$\theta = \varphi(d(X,Y)/\,\rho(X,Y) + \sigma) \tag{21}$$

Suppose the two contrastive spatial vectors are grayscale matrices X and Y of face images. Each matrix contains c pixels. Then, the disturbance φ can be calculated by:

$$\varphi = \begin{cases} \theta' \times (\rho(X,Y) + \sigma)/d(X,Y), & \mu(X,Y) < 0.078 \times c \\ 1, & 0.078 \times c \le \mu(X,Y) \le 0.156 \times c \\ 1/lg\times\big(m \times n - \mu(X,Y)\big), & \mu(X,Y) > 0.156 \times c \end{cases} \tag{22}$$

where, c is sample size; θ' is a preset threshold.

Structural similarity (SSIM) is a full reference image quality evaluation index. The traditional image quality metrics, such as mean squared error (MSE) and peak signal-to-noise ratio (PSNR), deviate from the actual visual perception of human eyes. By contrast, SSIM considers the visual features of human eyes, and adapts to the visual perception of humans. MSE and PSNR evaluates absolute errors, while SSIM, a perception-based metric, takes account of the fuzzy perceptual changes of image structure. The SSIM includes some phenomena related to perceptual changes, including brightness and contrast. The structural information refers to the internal dependence between pixels, especially the spatially close ones. The dependence carries important information about the visual perception of objects.

The SSIM ranges from 0 to 1. The greater the SSIM, the higher the similarity between two images. Based on the SSIM theory, the structural information is defined from the angle of image composition as an attribute reflecting the object structure in the scene, and independent of brightness and contrast, while distortion is modeled as the combination of brightness, contrast, and structure. Let X and Y be the reference image and the distorted image. Then, the following definitions can be established:

$$l\big(T_{i\times j}, T_{(i\pm1)\times(j\pm1)}\big) = (2u_X u_Y + C_1)/(u_X^2 + u_Y^2 + C_1) \tag{23}$$

$$c(X,Y) = (2\sigma_X\sigma_Y + C_2)/(\sigma_X^2 + \sigma_Y^2 + C_2) \tag{24}$$

$$s(X,Y) = (\sigma_{XY} + C_3)/(\sigma_X\sigma_Y + C_3) \tag{25}$$

where, u_X and u_Y are the mean of images X and Y, respectively, reflecting the brightness of each image; σ_X and σ_Y are the variances of images X and Y, respectively, reflecting the contrast of each image; $C_1, C_2,$ and C_3 are very small positive integers that prevent the denominator from being zero. Based on the above three information, the SSIM can be calculated by:

$$SSIM(X,Y) = [l(X,Y)]^{\alpha} \cdot [c(X,Y)]^{\beta} \cdot [s(X,Y)]^{\gamma} \tag{26}$$

where, $\alpha, \beta,$ and γ are the weights of different eigenvalues. If $\alpha = \beta = \gamma = 1, C_1 = (K_1 L)^2, C_2 = (K_2 L)^2, C_3 = C_2/2$, and $K_1 \ll 1, K_2 \ll 1$, with L being the dynamic range of the image, formula (26) can be simplified as:

$$SSIM(X,Y) = \frac{(2u_X u_Y + C_1)}{(2\sigma_X\sigma_Y + C_2)}/(u_X^2 + u_Y^2 + C_1)(\sigma_X^2 + \sigma_Y^2 + C_2) \tag{27}$$

To sum up, the FSMM mechanism can be expressed as:

$$FSMM(X,Y) = \frac{\varphi \times (d(X,Y)/\,\rho(X,Y) + \sigma) \times (2u_X u_Y + C_1) \times (2\sigma_X\sigma_Y + C_2)}{(u_X^2 + u_Y^2 + C_1) \times (\sigma_X^2 + \sigma_Y^2 + C_2)} \tag{28}$$

The innovation of RGP algorithm lies in merging similar sub-images into one region via region growing. When RGP and LAP have the same privacy budget, RGP can produce a smaller noise error during privacy protection of images. During image growing, the face image $X_{m\times n}$ is segmented into k sub-images of the same structure. Then, a sub-image T_{ij} is selected randomly as the seed point, and assigned a privacy budget of ε/k. Then, LAP is called to add a Laplace noise to the produce T'_{ij}. Whether an adjacent $T_{(i\pm1)(j\pm1)}$ needs to be merged into the growing region of the current seed point depends on whether the gap between T'_{ij} and $T_{(i\pm1)(j\pm1)}$ satisfies the preset threshold. If $FSMM\big(T'_{ij}, T_{(i\pm1)(j\pm1)}\big) \le Th$ (Th is the preset threshold), $T_{(i\pm1)(j\pm1)}$ will be replaced with T'_{ij} before publication. Once no more qualified sub-image can be merged with T_{ij} being the seed point, a new seed point will be selected randomly from the remaining sub-images, and the above process will be repeated until the entire image is fully grown.

Note that no privacy budget is consumed as $T_{(i\pm1)(j\pm1)}$ is replaced with T'_{ij}. Hence, the privacy budget assigned to $T_{(i\pm1)(j\pm1)}$ will be retained for use in subsequent operations.

To facilitate understanding, an example was provided to explain the implementation of RGP. The sub-image division is displayed in Figure 4: $X_{m\times n} = (T_{11}, T_{12}, T_{13}, \dots, T_{55})$. Out of the sub-images, a random sub-image is selected as the seed point, and assigned a privacy budget of the size $\varepsilon/25$. As shown in Figure 5, T'_{22} serves as the seed point, i.e., the disturbance obtained by adding Laplace noise to T_{22}. In this case, the state of T_{22} is labeled. After the seed point is selected, a chain will be created to temporarily store the seed points and candidate seed points.

Next, the seed point will be compared with each of the eight adjacent sub-images via region growing (Figure 6). If a sub-image is sufficiently similar and in line with the merging condition, replacement will be carried out, and the sub-image will be added to the temporary chain list as a candidate seed point. As shown in Figure 7, T_{23} is the only sub-image whose similarity with the seed point satisfies the preset threshold. Hence, T_{23} is replaced with T'_{22}, the current seed point is removed from the chain list, and the candidate seed point is selected as the new seed point. The above process is repeated until the temporary chain list is empty.

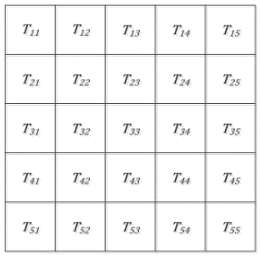

Figure 4. Image division

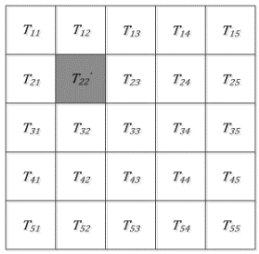

Figure 5. Seed point selection

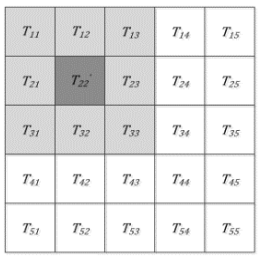

Figure 6. Traversing adjacent regions

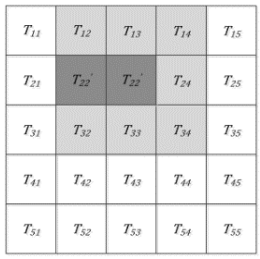

Figure 7. Replacement

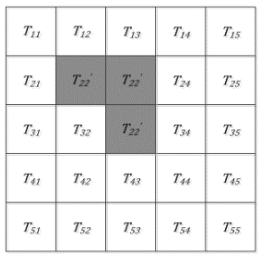

Figure 8. Result of local region growing

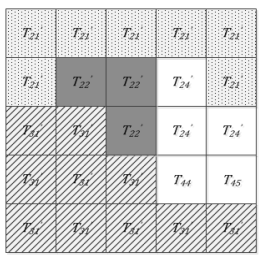

Figure 9. Result of global region growing

Figure 8 shows the labeled region after one region growing. A seed is selected from the unlabeled region, and the above steps are repeated, until all sub-images are labeled. Figure 9 shows the noisy grayscale matrix after four region growing operations.

The specific implementation of RGP is summarized as follows:

Algorithm 1: RGP

Inputs: Original image X, privacy budget ε, preset parameters a and b, and expectation for sub-image similarity Th

Output: Image X' satisfying differential privacy

1. Read original image X, and convert the image into a grayscale matrix $X_{m\times n}$ by formula (1).
2. Segment the grayscale matrix $X_{m\times n}$ into a set $T_1[a][b]$ of sub-images with the same structure, according to preset parameters a and b.
3. Create $T_2[a][b]=\{0\}$ to record whether the state of a submatrix is labeled during region growing.
4. Create a chain list *list* to temporarily store the current seed point and candidate seed points.
5. s=0;
6. $\varepsilon_{left}=\varepsilon$;
7. While(z<a*b)
8. if list_head == NULL
9. Randomly select $T_1[i][j]$ as the seed point
10. if $T_2[i][j]$=1 then
11. break;
12. else
13. i=head->x;
14. j=head->y;
15. p=head;
16. head=p->next;
17. free(p);
18. $T_2[i][j]=1$;
19. $T_1'[i][j] = T_1[i][j] + lap(\Delta Q \times (a * b - s)/\varepsilon_{left})$
20. $\varepsilon_{left} = \varepsilon_{left} - \varepsilon_{left}/(a * b - s)$
21. s=s+1;
22. p=list_head;
23. for m=i-1 to i+1 do
24. for n=j-1 to j+1 do
25. If(m>=1&&m<=a&&n>=1 &&n<=b&&$T_2[i][j]$!=1)
26. Use FSMM to compute the similarity θ between the current noisy seed point $T_1'[i][j]$ and $T_1[m][n]$.
27. If $\theta <= Th$
28. $T_1[m][n] = T_1'[i][j]$;
29. while(p->next!=NULL)
30. if(p->x==m&&p->y==n)
31. break;
32. p=p->next;
33. If p==NULL
34. Add $T_1[m][n]$ to the chain list.
35. Merge all the labeled submatrices into a noisy grayscale matrix $X_{m\times n}'$.
36. Convert $X_{m\times n}'$ into the privacy protected image X'.

RGP provides a way to protect the privacy of face images based on region growing and differential privacy. Lines 5-7

define two variables and set the termination condition (while circulation). The two variables are z and $\varepsilon_{\text{left}}$. The former ensures that, after the algorithm completes, each sub-image $T_1[i][j]$ will be converted into a noisy sub-image $T_1'[i][j]$, while $\varepsilon_{\text{left}}$ defines the initial value for the allocation of privacy budget ε. Lines 8-18 specifies the seed point selection of region growing. Specifically, Lines 8-11 judge whether $T_1[i][j]$ has been merged to a region. If it has been labeled, it is necessary to choose a new seed point. Lines 12-17 release the labeled seed point from the chain list, and select the next seed point from the candidates. Line 18 changes the state of the current seed point from unlabeled to labeled. Lines 19-21 assign a proper privacy budget and add a noise to the current seed point. Line 22 saves the current seed point to the chain list. Lines 23-25 demand that the adjacent region $T_1[m][n]$ of the seed point should not surpass the image boundary and the labeled state. Lines 26-28 use the FSMM mechanism to compute the similarity between an adjacent region and the seed point, and judge if the former meets the growth rule. If yes, $T_1[m][n]$ will be replaced with $T_1'[i][j]$. Lines 29-34 judge whether $T_1[m][n]$ that satisfies growth rule exists in the chain list. If not, this region will be added to the chain list, and become a candidate seed point.

Note that Lines 19-21 of RGP provide a dynamic allocation (DA) mechanism for privacy budget. Unlike the common binary allocation mechanism, DA offers a more reasonable solution to the privacy budget allocation during region growing, and reduces the influence of noise on the original data. The DA mechanism can be described as follows:

Suppose each region growing consumes a privacy budget of $\varepsilon_i = \{\varepsilon_1, \varepsilon_2, \varepsilon_3, \dots, \varepsilon_m\}$, each merged region contains $s_i = \{s_1, s_2, s_3, \dots, s_m\}$ sub-images, and $s_1 + s_2 + s_3 + \cdots + s_m = k$, with $k = a \times b$ being the total number of sub-images. Let $\varepsilon_i^{\text{left}}$ be the residual privacy budget after the completion of state i. Then, the ε_i and $\varepsilon_{\text{ileft}}$ in any state under the DA mechanism can be expressed as:

$$\varepsilon_i = \varepsilon_{i-1}^{left} \Big/ \left(k - \sum_{j=1}^{i-1} s_j\right) \tag{29}$$

$$\varepsilon_i^{left} = \varepsilon_{i-1}^{left} - \varepsilon_i \tag{30}$$

Theorem 3. In RGP, the consumption of privacy budget will not surpass ε, i.e., $\varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \cdots + \varepsilon_m \le \varepsilon$.

Proof: Formula (29) shows that, when $k - \sum_{j=1}^{m-1} s_j = 1$, $\varepsilon_{m-1\text{left}} = \varepsilon_m$. Then,$\varepsilon_{\text{mleft}} = 0$. In this case, we have:

$$\varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \cdots + \varepsilon_m = \varepsilon$$

When $k - \sum_{j=1}^{m-1} s_j > 1$, $\varepsilon_{m-1\text{left}}/\varepsilon_m > 1$. Then, $\varepsilon_{\text{mleft}} > 0$, In this case, we have:

$$\varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \cdots + \varepsilon_m < \varepsilon$$

Q.E.D.

Theorem 4. RGP satisfies ε- differential privacy.

Proof: According to Theorem 3:

$$\varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \cdots + \varepsilon_m \le \varepsilon$$

where, $\varepsilon_i (1 \le i \le m)$ is the privacy budget consumed under each state. According to the differential privacy-parallel combination property (Property 2), RGP satisfies ε - differential privacy.

Q.E.D.

Theorem 5. The error generated by RGP is no greater than that generated by LAP, i.e.:

$$Error(RGP) \le Error(LAP)$$

Proof: The DA mechanism of RGP consumes a privacy budget of $\varepsilon_i = \{\varepsilon_1, \varepsilon_2, \varepsilon_3, \dots, \varepsilon_m\}$ each time. The number of submatrices affected by $\varepsilon_i = \{\varepsilon_1, \varepsilon_2, \varepsilon_3, \dots, \varepsilon_m\}$ can be described by $s_i = \{s_1, s_2, s_3, \dots, s_m\}$, $s_1 + s_2 + s_3 + \cdots + s_m = k$. The Laplace mechanism of LAP consumes a privacy budget ofε/k, $\sum_{i=1}^{k} s_i' = k$ ($s_1' = s_2' = s_3' = \cdots = s_k' = 1$) each time.

Firstly, the maximum possible error of DA mechanism is calculated, revealing that $s_1 = s_2 = s_3 = \cdots s_m = 1$, i.e., $m = k$. Thus, we have:

$$Error(RGP)_{max} = \big(lap(\Delta Q/\varepsilon_1) + lap(\Delta Q/\varepsilon_2) + lap(\Delta Q/\varepsilon_3) + \cdots lap(\Delta Q/\varepsilon_m)\big)_{max} = \sum_{i=1}^{k} lap(\Delta Q/\varepsilon_i) = k \times lap\big(\Delta Q/(\varepsilon/k)\big) = Error(LAP)$$

The other situations can be judged by formula (30). After the completion of state i, the residual privacy budget of DA mechanism is $\varepsilon_{\text{ileft}}$, while that of Laplace mechanism is $\varepsilon_{\text{ileft}}'$. Since DA mechanism allows an $s_i > 1$, and $s_1 = s_2 = s_3 = \cdots s_{i-1} = 1 (1 < i < m)$, we have:

$$\varepsilon_{ileft} - \varepsilon_{ileft}' = \left(\varepsilon - \varepsilon \times \frac{k-(i-1+s_i)}{k}\right) - \left(\varepsilon - \varepsilon \times \frac{k-i}{k}\right) = \varepsilon \times \left(\frac{k-i}{k} - \frac{k-(i-1+s_i)}{k}\right) = \varepsilon \times \frac{s_i - 1}{k}$$

Since $s_i > 1$, $\varepsilon_{\text{ileft}} - \varepsilon_{\text{ileft}}' > 0$. Thus, $(\varepsilon_{i+1}, \varepsilon_{i+2}, \varepsilon_{i+3}, \dots, \varepsilon_m) > \varepsilon/k$. It can be seen that $\text{Error(RGP)} < \text{Error(LAP)}$.

Q.E.D.

3.3 Atypical region growing publication (ARGP)

If the RGP is applied to an image, the merged region through one region growing with T_{11} as the seed point can be illustrated in Figure 10 (the dark region). From the perspective of the entire image, i.e., removing the limitation on region growing direction, the similarity θ between T_{11}' and each unlabeled sub-image is calculated, and all the qualified sub-images are merged together. But the merged region may not be as expected (Figure 11). Obviously, the current merged region is not continuous in 2D space (RGP cannot yield such a merged region), but every sub-image of the merged region satisfies θ <= Th.

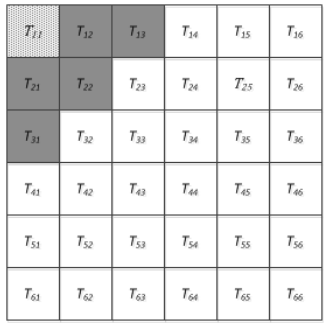

Figure 10. Region growing direction estimated by RGP

To find more qualified sub-images for the seed point without damaging ε-differential privacy, this paper integrates atypical region growing with differential privacy into a novel privacy protection approach for image publication, denoted as ARGP. Different from traditional region growth strategy, the ARGP does not restrict the growing direction within adjacent sub-images, but searches for qualified sub-images among all unlabeled sub-images, using the exponential mechanism. The specific steps of the ARGP are as follows:

Algorithm 2. ARGP

Inputs: Original image X, privacy budget ε, preset parameters a and b, and expectation for sub-image similarity Th

Output: Image X′ satisfying differential privacy

1. Read original image X, and convert the image into a grayscale matrix $X_{m\times n}$ by formula (1).
2. Segment the grayscale matrix $X_{m\times n}$ into a set $T_1[a][b]$ of sub-images with the same structure, according to preset parameters a and b.
3. Create $T_2[a][b]=\{0\}$ to record whether the state of a submatrix is labeled during region growing.
4. $\varepsilon=\varepsilon_1+\varepsilon_2$;
5. s=0;
6. $\varepsilon_{left}=\varepsilon_2$;
7. While(w<a*b)
8. If i==0&&j==0
9. Randomly select $T_1[i][j]$ as the seed point
10. endif
11. $T_2[i][j]=1$;
12. $T_1'[i][j] = T_1[i][j] + lap(\Delta Q \times (a * b - s)/\varepsilon_{left})$;
13. $\varepsilon_{left} = \varepsilon_{left} - \varepsilon_{left}/(a * b - s)$
14. s=s+1;
15. Find all unlabeled sub-images $T_1[a][b]$.
16. Realize the region merging in the traditional direction of region growing of RGP.
17. Find all unlabeled sub-images $T_1[a][b]$ again.
18. Use FSMM to compute the similarity θ between $T_1'[i][j]$ and each $T_1[a][b]$.
19. $\Delta Q=\frac{1}{\theta+\sigma}$;
20. Select one $T_1[a][b]$ at the probability $P \propto \exp\left(\frac{\varepsilon_1 \Delta Q}{2\Delta u}\right)$, using the exponential mechanism.
21. Use FSMM to recalculate the similarity θ between the noisy current seed point $T_1'[i][j]$ and the selected $T_1[a][b]$.
22. If $\theta <= Th$
23. $T_1[a][b] = T_1'[i][j]$;
24. i=m;
25. j=n;
26. else
27. i=0;
28. j=0;
29. Merge all the labeled submatrices into a noisy grayscale matrix $X_{m\times n}'$.
30. Convert $X_{m\times n}'$ into the privacy protected image X′.

In ARGP, Line 4 divides the privacy budget ε into two parts: a part ε_1 for exponential mechanism, and a part ε_2 for noise addition. Lines 5-7 define two variables and set the termination condition (while circulation). The two variables are w and ε_{left}. The former ensures that, after the algorithm completes, each sub-image $T_1[i][j]$ will be converted into a noisy sub-image $T_1'[i][j]$, while ε_{left} defines the initial value for the allocation of privacy budget ε. Lines 8-14 specifies that, if there is no seed point, an unlabeled sub-image will be selected randomly as the seed point; then, the seed point will be labeled, and added a noise by DA mechanism. Lines 15-16 realize the RGP. Lines 17-20 select a sub-image by the exponential mechanism. Specifically, Line 17 defines the scoring function of exponential mechanism. ARGP intends to select the sub-image more similar to the seed point at a larger probability. Σ is a very small positive number that prevents the denominator from being zero. In Line 18, Δu is the sensitivity of the scoring function. Since adding/removing one record only affects one count of θ, Δu=1. Lines 21-28 use the FSMM mechanism to compute the similarity between the seed point and the selected $T_1[m][n]$. If the result satisfies the preset threshold, assign the value of $T_1'[i][j]$ to $T_1[m][n]$, and set it as the new seed point. Otherwise, select a new seed point, and repeat the above process until all sub-images become noisy.

For ARGP, the exponential mechanism does not necessarily lead to the needed optimal solution, but capable of obtaining the optimal or near optimal solution at a high probability. Table 1 shows the test results on the selection probability of the exponential mechanism in ARGP. The test aims to select the minimum of a set of random numbers. Column 1 reports the similarity θ between the seed point and sub-regions. Column 2 reports the scoring function (σ=0.02). The other four columns report the section probabilities of each sub-region at different ε values, and the sum of each column equals or approximates 1 (the error induced by indivisibility). If ε=0, the level of privacy protection is maximized, and all sub-regions are selected at the same probability. With the growth of ε, the selection probability increases with the value of the scoring function ΔQ; the inverse is also true.

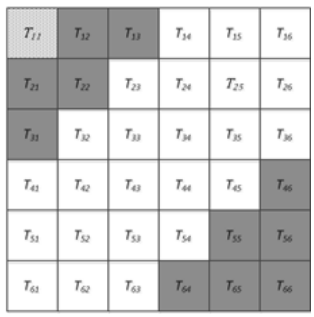

Figure 11. Region growing direction estimated by ARGP

Table 1. Exponential mechanism- minimum selection probability test

θ	ΔQ	ε=0	ε=0.05	ε=0.1	ε=0.5
15	0.0666	14.29%	2.15%	0.21%	0.20%
62	0.0161	14.29%	0.73%	0.00%	0.00%
28	0.0357	14.29%	1.01%	0.10%	0.00%
7	0.1425	14.29%	14.50%	6.12%	0.52%
8	0.1247	14.29%	13.21%	6.19%	0.51%
1	0.9804	14.29%	67.56%	87.32%	98.77%
37	0.0270	14.29%	0.86%	0.06%	0.00%

Theorem 6. ARGP satisfies ε-differential privacy.

Proof: ARGP divides the privacy budget ε into two parts: $\varepsilon=\varepsilon_1+\varepsilon_2$, where ε_1 is used to select unlabeled sub-region T by the exponential mechanism; the selection probability is positively correlated with $\exp\left(\frac{\varepsilon_1 \Delta Q}{2\Delta u}\right)$:

$$\Pr[M(X,\Delta Q)=T]=\frac{\exp\left(\frac{\varepsilon_1\Delta Q(X,T)}{2\Delta u}\right)}{\sum_{T'\in O}\exp\left(\frac{\varepsilon_1\Delta Q(X,T')}{2\Delta u}\right)} \quad (31)$$

For the given X and its adjacent region X′, the following can be derived from formula (31) for any $T \in O$:

$$\frac{Pr[M(X,\Delta Q)=T]}{Pr[M(X',\Delta Q)=T]}=\frac{\frac{exp\left(\frac{\varepsilon_1\Delta Q(X,T)}{2\Delta u}\right)}{\sum_{T'\in O}exp\left(\frac{\varepsilon_1\Delta Q(X,T')}{2\Delta u}\right)}}{\frac{exp\left(\frac{\varepsilon_1\Delta Q(X',T)}{2\Delta u}\right)}{\sum_{T'\in O}exp\left(\frac{\varepsilon_1\Delta Q(X',T')}{2\Delta u}\right)}}=\left(\frac{exp\left(\frac{\varepsilon_1\Delta Q(X,T)}{2\Delta u}\right)}{exp\left(\frac{\varepsilon_1\Delta Q(X',T)}{2\Delta u}\right)}\right)\times$$
$$\left(\frac{\sum_{T'\in O}exp\left(\frac{\varepsilon_1\Delta Q(X',T')}{2\Delta u}\right)}{\sum_{T'\in O}exp\left(\frac{\varepsilon_1\Delta Q(X,T')}{2\Delta u}\right)}\right)=exp\left(\frac{\varepsilon_1\left(\varepsilon_1\Delta Q(X,T)-\varepsilon_1\Delta Q(X',T)\right)}{2\Delta u}\right)\times$$
$$\left(\frac{\sum_{T'\in O}exp\left(\frac{\varepsilon_1\Delta Q(X',T')}{2\Delta u}\right)}{\sum_{T'\in O}exp\left(\frac{\varepsilon_1\Delta Q(X,T')}{2\Delta u}\right)}\right)\le exp\left(\frac{\varepsilon_1}{2}\right)\times$$
$$\left(\frac{\sum_{T'\in O}exp\left(\frac{\varepsilon_1}{2}\right)\times exp\left(\frac{\varepsilon_1\Delta Q(X,T')}{2\Delta u}\right)}{\sum_{T'\in O}exp\left(\frac{\varepsilon_1\Delta Q(X,T')}{2\Delta u}\right)}\right)\le exp\left(\frac{\varepsilon_1}{2}\right)\times exp\left(\frac{\varepsilon_1}{2}\right)\times$$
$$\left(\frac{\sum_{T'\in O}exp\left(\frac{\varepsilon_1\Delta Q(X,T')}{2\Delta u}\right)}{\sum_{T'\in O}exp\left(\frac{\varepsilon_1\Delta Q(X,T')}{2\Delta u}\right)}\right)=exp(\varepsilon_1)$$

Hence, the exponential-based selection of unlabeled sub-region of ARGP satisfies ε_1-differential privacy. From the proof of Theorem 4, it can be learned that the noise addition of ARGP satisfies ε_2-differential privacy. Overall, ARGP satisfies ε- differential privacy in the whole process.

Q.E.D.

Theorem 7. When an image is divided on equal conditions, the number z_j of sub-images to be merged to a seed point in ARGP must be greater than or equal to the number s_i of sub-images to be merged to the same seed point in RGP, under ideal conditions, that is, $\mathrm{Error(ARGP)} \le \mathrm{Error(RGP)}$.

Proof: Theorem 7 can be proved by contradiction. It is assumed that $z_j < s_i$.

In ARGP, there exists a sub-image T_0, whose merged region is set A, i.e., $T_0 \in A$, with z_j being the number of elements in set A. In RGP, the merged region of the same sub-image T_0 is set B, i.e., $T_0 \in B$, with s_i being the number of elements in set B. If $z_j < s_i$ holds, then there must exist a sub-image T_x satisfying $T_x \notin A$ in ARGP and $T_x \in B$ in RGP.

From the growing direction and seed point selection of the two algorithms, if $T_x \in B$, then there must exist $T_x \in A$. Thus, $z_j < s_i$ is invalid.

Q.E.D.

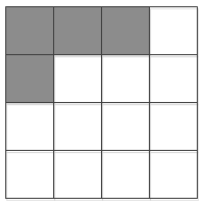

Figure 12. $z_j = s_i$

The conclusion of Theorem 7 can be further analyzed by dividing $z_j \ge s_i$ into $z_j = s_i$ and $z_j > s_i$. Suppose there is an image will produce δ merged regions by RGP or ARGP (Figure 12; the merged regions are in dark color). If the exponential mechanism cannot obtain the qualified sub-images for merging, then any $z_j = s_i$. From the proof of Theorem 5, we have:

$$\begin{cases} Error(ARGP)_{\max} \\ = \mathrm{Error}(RGP)_{\max} & \delta = 0 \\ = \mathrm{Error(Lap)}, \\ \mathrm{Error}(ARGP) & 1 \le \delta \le k/2 \\ = \mathrm{Error}(RGP) & or \\ < \mathrm{Error(Lap)}, & -1 \le \delta \le (a*b)/2 \end{cases}$$

As shown in Figure 13, if the exponential mechanism can obtain any qualified sub-images, which are not connected with the current merged region, then $z_j > s_i$. It can be derived from formula (30) that:

$$\varepsilon_{RGPleft}-\varepsilon_{ARGPleft}=\left(\varepsilon-\varepsilon\times\frac{k-(\gamma-1+s_i)}{k}\right)-\left(\varepsilon-\varepsilon\times\frac{a*b-(\gamma-1+z_j)}{a*b}\right)=\varepsilon*\left(\frac{a*b-(\gamma-1+z_j)}{a*b}-\frac{k-(\gamma-1+s_i)}{k}\right)=\varepsilon\times\frac{s_i-z_j}{k}$$

Since $z_j > s_i$, $\varepsilon_{\mathrm{RGPleft}}-\varepsilon_{\mathrm{ARGPleft}} < 0$. Thus, $\mathrm{Error(ARGP)} < \mathrm{Error(RGP)}$.

Q.E.D.

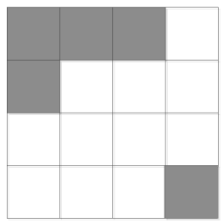

Figure 13. $z_j > s_i$

4. EXPERIMENTS AND RESULTS ANALYSIS

4.1 Experiments

Among the various kinds of images, face images have the most representative sensitivity information. To verify its feasibility, our algorithm was tested on LENA.JPG (size: 512*512). During the execution of the algorithm, the original image was split into 4,096 sub-images of the size 8*8. The following figures present the test results of different algorithms under the same conditions: Figure 14 shows the original image; Figure 15 shows the results of direct addition of Laplace noise to the original image; Figure 16 shows the result obtained by RGP; Figure 17 shows the result obtained by ARGP. Apparently, Figures 16 and 17 are closer to the original image than Figure 15, and the result of ARGP is better than that of LPA and RGP.

Figure 14. Original image

Figure 15. Result of LAP

Figure 16. Result of RGP

Figure 17. Result of ARGP

Now, a preliminary conclusion can be drawn: RGP and ARGP are superior to LAP. But the conclusion is merely obtained by observing the images on test results. It might be one-sided or inaccurate. To further validate the conclusion, the four images were converted into grayscale histograms and compared once more. Grayscale histogram demonstrates the relationship between the occurrence frequency of pixels on each grayscale and grayscale. Although it does not reflect the specific distribution of image pixels, the grayscale histogram statistically showcases the similarity between different images.

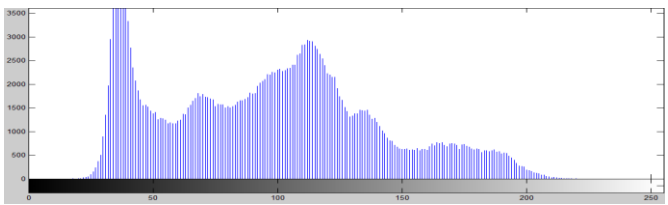

Figure 18. Grayscale histogram of the original image

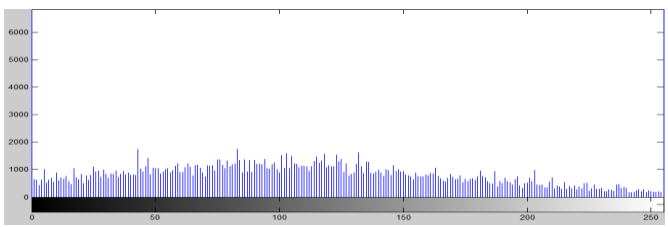

Figure 19. Grayscale histogram of LAP result

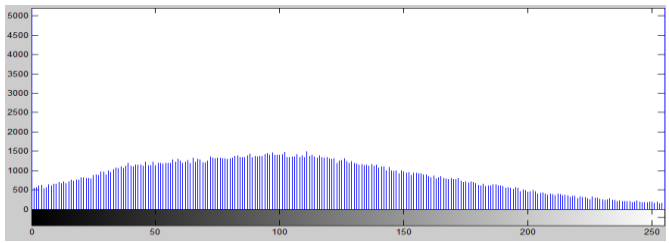

Figure 20. Grayscale histogram of RGP result

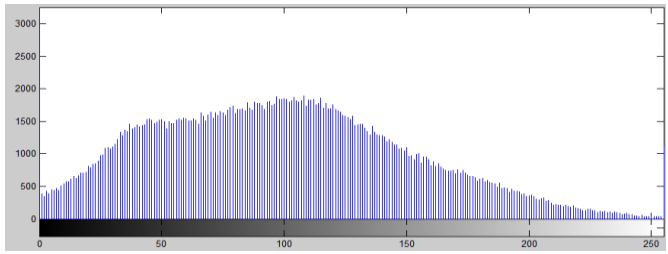

Figure 21. Grayscale histogram of ARGP result

As shown in Figures 18-21, under the same conditions, the histogram of RGP has similar distribution as that of LAP, but differs significantly from that of the original image. The pixel distribution of ARGP's histogram is basically the same as that of the original image. Hence, ARGP has the least noise disturbance to the image, and achieves the highest availability.

4.2 Result analysis

Table 2. ORL & Precision

Precision	LAP	SWP	**RGP**	Sort-SWP	**ARGP**
ε=1	6.5%	39.3%	**47.4%**	53.9%	**54.6%**
ε=ln7	8.7%	42.4%	**56.5%**	69.0%	**72.3%**
ε=ln20	11.4%	47.3%	**63.6 %**	71.9%	**73.2%**
ε=5	16.7%	62.7%	**70.1%**	82.8%	**83.6%**
ε=10	18.6%	68.7%	**73.6%**	85.1%	**85.7%**

Table 3. ORL & Recall

Recall	LAP	SWP	**RGP**	Sort-SWP	**ARGP**
ε=1	71.3%	78.0%	**80.6%**	83.6%	**82.7%**
ε=ln7	73.2%	75.0%	**81.7%**	81.2%	**83.1%**
ε=ln20	74.7%	83.8%	**82.1%**	84.0%	**83.9%**
ε=5	77.9%	76.3%	**84.6%**	86.1%	**88.0%**
ε=10	80.1%	82.2%	**86.6%**	88.1%	**88.3%**

Table 4. ORL & F1-score

F1-score	LAP	SWP	**RGP**	Sort-SWP	**ARGP**
ε=1	11.9%	52.3%	**59.7%**	65.5%	**65.8%**
ε=ln7	15.5%	54.2%	**66.8%**	74.6%	**77.3%**
ε=ln20	19.8%	60.5%	**71.7%**	77.5%	**78.2%**
ε=5	27.5%	68.8%	**76.7%**	84.4%	**85.7%**
ε=10	30.2%	74.8%	**79.6%**	86.6%	**87.0%**

Table 5. YALE & Precision

Precision	LAP	SWP	**RGP**	Sort-SWP	**ARGP**
ε=1	4.1%	27.3%	**33.2%**	44.5%	**55.3%**
ε=ln7	6.5%	31.1%	**37.8%**	56.3%	**60.2%**
ε=ln20	9.6%	39.0%	**45.2%**	62.7%	**66.9%**
ε=5	13.0%	59.3%	**60.1%**	75.9%	**77.3%**
ε=5	18.6%	63.5%	**68.5%**	77.4%	**78.9%**

Table 6. YALE & Recall

Recall	LAP	SWP	RGP	Sort-SWP	ARGP
ε=1	77.3%	77.5%	**77.4%**	84.2%	**84.0%**
ε=ln7	77.6%	77.7%	**79.1%**	81.7%	**83.1%**
ε=ln20	79. 4%	82.6%	**85.8%**	86.8%	**87.5%**
ε=5	75.4%	82.4%	**86.2%**	85.7%	**88.0%**
ε=10	77.3%	85.2%	**88.7%**	89.2%	**90.0%**

Table 7. YALE & F1-score

F1-sroce	LAP	SWP	RGP	Sort-SWP	ARGP
ε=1	7.8%	40.4%	**46.5%**	58.2%	**66.7%**
ε=ln7	12.0%	44.4%	**51.2%**	67.7 %	**69.8%**
ε=ln20	17.1%	53.0%	**59.2%**	72.8%	**75.8%**
ε=4	19.6%	59.3%	**67.0%**	75.6%	**78.9%**
ε=10	30.0%	72.8%	**77.3%**	82.9%	**84.1%**

Table 8. IMM & Precision

Precision	LAP	SWP	RGP	Sort-SWP	ARGP
ε=1	3.9%	45.6%	**50.1%**	62.6%	**65.4%**
ε=ln7	6.5%	51.2%	**63.2%**	72.3%	**76.5%**
ε=ln20	8.6%	57.8%	**69.1%**	75.3%	**82.1%**
ε=5	10.7%	68.6%	**75.8%**	84.0%	**87.5%**
ε=10	15.2%	75.6%	**79.6%**	88.9%	**91.2%**

Table 9. IMM & Recall

Recall	LAP	SWP	RGP	Sort-SWP	ARGP
ε=1	54.6%	81.2%	**83.7%**	84.1%	**85.6%**
ε=ln7	58.2%	80.7%	**82.7%**	84.5%	**85.9%**
ε=ln20	63.8%	85.4%	**86.3%**	88.7%	**89.0%**
ε=5	66.7%	78.9%	**83.6%**	87.8%	**90.1%**
ε=10	70.1%	83.2%	**85.9%**	88.0%	**91.2%**

Table 10. IMM & F1-score

F1-sroce	LAP	SWP	RGP	Sort-SWP	ARGP
ε=1	7.3%	58.4%	**62.7%**	71.8%	**74.1%**
ε=ln7	11.7%	62.7%	**71.6%**	78.0%	**80.9%**
ε=ln20	15.2%	68.9%	**76.7%**	81.5%	**85.4%**
ε=5	18.4%	73.4%	**79.5%**	85.9%	**88.8%**
ε=10	25.0%	79.2%	**82.6%**	88.4%	**91.2%**

To verify the feasibility of our algorithm, experiments were carried out on ORL, YALE, and IMM face databases, under the environment of Intel® Core i9-9900K CPU @ 3.60 GHz, 32G memory, GTX 21080TI GPU, and Windows 10 operating system. During the experiments, a facial recognition method based on improved AlexNet, a convolutional neural network (CNN), was adopted. This approach has a simpler structure and fewer parameters than AlexNet, and thus saves lots of training time, laying the basis for rapid prediction. The sub-image size was set to 4*4 for the experiment on ORL database, 5*5 for that on YALE database, and 8*8 for that on IMM database. The privacy budget ε was set to 1, ln7, ln20, 5, and 10, respectively. The test metrics are precision, recall, and F1-score. The test results are listed in Tables 2-10.

For differential privacy, the noise level is an important metric of algorithm performance. Under the same conditions, fewer noise means higher feasibility. Of course, noise level is only one of the indices of images, a special information carrier. Our algorithm needs to divide the original image into multiple sub-images before subsequent computing. The sub-image size affects the division results, which in turn influence the total noise generated by the algorithm. In addition to LAP, RGP, and ARGP, XXX's sliding window publication (SWP) algorithm [1] and sort-SWP algorithm are compared in our experiments. Note that SWP and sort-SWP are realized based on 1D data flows. During image division, the two algorithms can only compute 1D sub-images.

The image size varies between face image databases: 92*112 in ORL, 100*100 in YALE, and 480*640 in IMM. The image division on each database must respect the size of original images. Thus, it is impossible to unify the sub-image size across the databases. The experimental results are displayed in Figures 22-24.

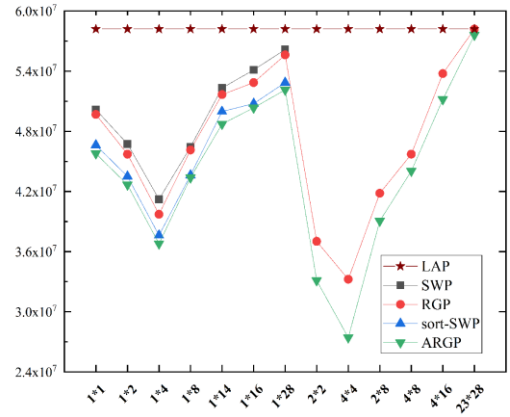

Figure 22. Influence of ORL & sub-image size on noise

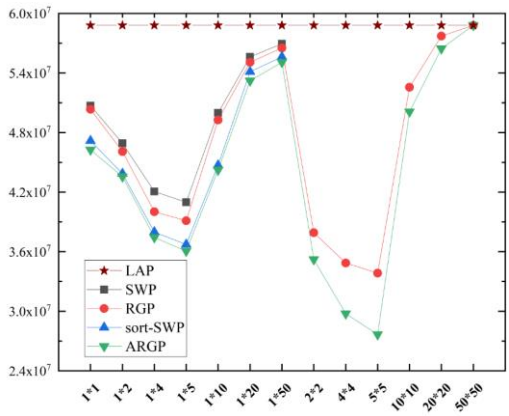

Figure 23. Influence of YALE & sub-image size on noise

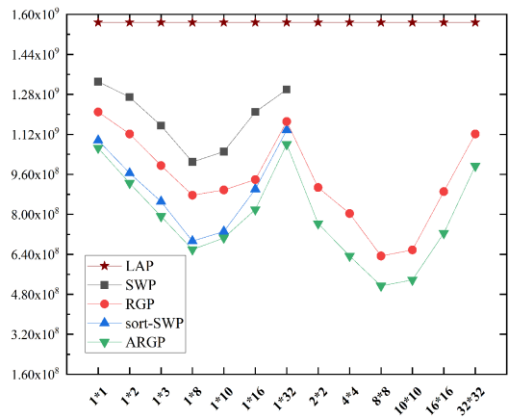

Figure 24. Influence of IMM & sub-image size on noise

Experimental results show that the total noise generated by an algorithm change with the sub-image size. When the sub-images are too large or too small, the images after privacy

protection will have a low availability. The optimal sub-image size is 4*4 for ORL, 5*5 for YALE, and 8*8 for IMM. Thus, it can be derived that the optimal sub-image size is related to the size of the original image. Drawing on the results in Tables 2-10 and Figures 22-24, it was found that image quality affects the final operation results of the algorithms. The better the image quality, the more in line the results are with our expectation.

5. CONCLUSIONS

To solve the privacy protection of face image publication, this paper combines regional growing technique with the Laplace mechanism of differential privacy to add noise to the original image, thereby protecting the sensitive information in face images. Compared with the LAP algorithm, which directly adds Laplace noise to the image, RGP and ARGP can effectively reduce the influence of noise on the protected image, and improve the feasibility of privacy protection on images. Moreover, this paper presents a novel region growing rule: FSMM, and discusses the influence of sub-image size on algorithm results. These conclusions provide an effective reference for other researchers.

It is worth noting that our differential privacy protection approach for face image publication is realized by global noise addition. However, the sensitive information of a face image mostly exists in specific regions (e.g., facial contours, eyes, eyebrows, mouth, and noise). The future work will try to pinpoint the locations of sensitive information, and add noise to these places, aiming to further reduce the noise influence, and enhance the availability of face images after privacy protection.

ACKNOWLEDGMENT

This work is supported by National Natural Science Foundation of China (Grant No.: 61672179), Natural Science Foundation, Heilongjiang Province, China (Grant No.: LH2021D022, 2019F004), and Fundamental Research Funds, Heilongjiang Provincial Education Department, China (Grant No.: 135309457).

REFERENCES

[1] Liu, C., Yang, J., Zhao, W., Zhang, Y., Li, J., Mu, C. (2021). Face image publication based on differential privacy. Wireless Communications and Mobile Computing, 2021(9): 1-20. https://doi.org/10.1155/2021/6680701

[2] Fung, B.C., Wang, K., Philip, S.Y. (2007). Anonymizing classification data for privacy preservation. IEEE Transactions on Knowledge and Data Engineering, 19(5): 711-725. https://doi.org/10.1109/TKDE.2007.1015

[3] Xiao, X., Tao, Y. (2006). Anatomy: Simple and effective privacy preservation. In Proceedings of the 32nd International Conference on Very Large Data Bases, pp. 139-150.

[4] Li, T., Li, N., Zhang, J., Molloy, I. (2010). Slicing: A new approach for privacy preserving data publishing. IEEE Transactions on Knowledge and Data Engineering, 24(3): 561-574. https://doi.org/10.1109/TKDE.2010.236

[5] Terrovitis, M., Liagouris, J., Mamoulis, N., Skiadopoulos, S. (2012). Privacy preservation by disassociation. Proceedings of the VLDB Endowment, 5: 944-955. https://doi.org/10.14778/2336664.2336668

[6] Sweeney, L. (2002). K-anonymity: A model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 10(5): 557-570. https://doi.org/10.1142/S0218488502001648

[7] Dwork, C. (2006). Differential privacy. In International Colloquium on Automata, Languages, and Programming. Springer, Berlin, Heidelberg, 1-12. https://doi.org/10.1007/11787006_1

[8] Dwork, C. (2008). Differential privacy: A survey of results. In International Conference on Theory and Applications of Models of Computation. Springer, Berlin, Heidelberg, pp. 1-19. https://doi.org/10.1007/978-3-540-79228-4_1

[9] Dwork, C. (2009). The differential privacy frontier. In Theory of Cryptography Conference, Springer, Berlin, Heidelberg, 496-502. https://doi.org/10.1007/978-3-642-00457-5_36

[10] Dwork, C., Lei, J. (2009). Differential privacy and robust statistics. In Proceedings of the forty-first annual ACM symposium on Theory of computing, Bethesda, MD, USA, pp. 371-380. https://doi.org/10.1145/1536414.1536466

[11] Dwork, C. (2010). Differential privacy in new settings. In Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms, Society for Industrial and Applied Mathematics, pp. 174-183. https://doi.org/10.1137/1.9781611973075.16

[12] Dwork, C. (2011). The promise of differential privacy a tutorial on algorithmic techniques. In 2011 IEEE 52nd Annual Symposium on Foundations of Computer Science, pp. 1-2. https://doi.org/10.1109/FOCS.2011.88

[13] Dwork, C., McSherry, F., Nissim, K., Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. In Theory of Cryptography Conference, Springer, Berlin, Heidelberg, pp. 265-284. https://doi.org/10.1007/11681878_14

[14] McSherry, F., Talwar, K. (2007). Mechanism design via differential privacy. In 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07), pp. 94-103. https://doi.org/10.1109/FOCS.2007.66

[15] McSherry F. (2010). Privacy integrated queries: An extensible platform for privacy-preserving data analysis. Communications of the ACM, 53(9): 89-97. https://doi.org/10.1145/1810891.1810916

[16] Roth, A., Roughgarden, T. (2010). Interactive privacy via the median mechanism. In Proceedings of the Forty-Second ACM Symposium on Theory of Computing, pp. 765-774. https://doi.org/10.1145/1806689.1806794

[17] Hardt, M., Rothblum, G.N. (2010). A multiplicative weights mechanism for privacy-preserving data analysis. In 2010 IEEE 51st Annual Symposium on Foundations of Computer Science, pp. 61-70. https://doi.org/10.1109/FOCS.2010.85

[18] Gupta, A., Roth, A., Ullman, J. (2012). Iterative constructions and private data release. In Theory of Cryptography Conference, Springer, Berlin, Heidelberg, pp. 339-356. https://doi.org/10.1007/978-3-642-28914-9_19

[19] Fan, L., Xiong, L. (2012). Real-time aggregate monitoring with differential privacy. In Proceedings of

the 21st ACM International Conference on Information and Knowledge Management, pp. 2169-2173. https://doi.org/10.1145/2396761.2398595

[20] Kellaris, G., Papadopoulos, S., Xiao, X., Papadias, D. (2014). Differentially private event sequences over infinite streams. Proceedings of the VLDB Endowment, 7(12): 1155-1166. https://doi.org/10.14778/2732977.2732989

[21] Xiao, X., Wang, G., Gehrke, J. (2010). Differential privacy via wavelet transforms. IEEE Transactions on Knowledge and Data Engineering, 23(8): 1200-1214. https://doi.org/10.1109/TKDE.2010.247

[22] Xu, J., Zhang, Z., Xiao, X., Yang, Y., Yu, G., Winslett, M. (2013). Differentially private histogram publication. The VLDB Journal, 22(6): 797-822. https://doi.org/10.1007/s00778-013-0309-y

[23] Li, C., Miklau, G., Hay, M., McGregor, A., Rastogi, V. (2015). The matrix mechanism: optimizing linear counting queries under differential privacy. The VLDB Journal, 24(6): 757-781. https://doi.org/10.1007/s00778-015-0398-x

[24] Li, C., Hay, M., Miklau, G., Wang, Y. (2014). A data- and workload-aware algorithm for range queries under differential privacy. Proceedings of the VLDB Endowment, 7(5): 341-352. https://doi.org/10.14778/2732269.2732271

[25] Zhang, X.J., Fu, C.C., Meng, X.F. (2018). Facial image publication with differential privacy. Journal of Image and Graphics, 23(9): 1305-1315.

[26] Nissim, K., Raskhodnikova, S., Smith, A. (2007). Smooth sensitivity and sampling in private data analysis. In Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing, pp. 5-84. https://doi.org/10.1145/1250790.1250803

[27] Adams, R., Bischof, L. (1994). Seeded region growing. IEEE Transactions on Pattern Analysis and Machine Intelligence, 16(6): 641-647. https://doi.org/10.1109/34.295913

[28] Matalas, L., Benjamin, R., Kitney, R. (1997). An edge detection technique using the facet model and parameterized relaxation labeling. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(4): 328-341. https://doi.org/10.1109/34.588006