Local Privacy Protection for Sensitive Areas in Multiface Images

Chao Liu, Jing Yang, Xuan Zhang, Yining Zhang, Weinan Zhao, Fengjuan Miao, and Yukun Shao

1College of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China
2College of Communication and Electronic Engineering, Qiqihar University, Qiqihar 161000, China
3Information Center, First Affiliated Hospital of Qiqihar Medical College, Qiqihar 161000, China
4Department of Computer Engineering, Daqing Vocational College, Daqing 163000, China

Correspondence should be addressed to Jing Yang; yangjing@hrbeu.edu.cn

Received 6 January 2022; Revised 26 January 2022; Accepted 28 January 2022; Published 15 March 2022

Academic Editor: Daqing Gong

Copyright © 2022 Chao Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The privacy protection for face images aims to prevent attackers from accurately identifying target persons through face recognition. Inspired by goal-driven reasoning (reverse reasoning), this paper designs a goal-driven algorithm of local privacy protection for sensitive areas in multiface images (face areas) under the interactive framework of face recognition algorithm, regional growth, and differential privacy. The designed algorithm, named privacy protection for sensitive areas (PPSA), is realized in the following manner: Firstly, the multitask cascaded convolutional network (MTCNN) was adopted to recognize the region and landmark of each face. If the landmark overlaps a subgraph divided from the original image, the subgraph will be taken as the seed for regional growth in the face area, following the growth criterion of the fusion similarity measurement mechanism (FSMM). Different from single-face privacy protection, multiface privacy protection needs to deal with an unknown number of faces. Thus, the allocation of the privacy budget $\varepsilon$ directly affects the operation effect of the PPSA algorithm. In our scheme, the total privacy budget $\varepsilon$ is divided into two parts: $\varepsilon_{1}$ and $\varepsilon_{2}$. The former is evenly allocated to each seed, according to the estimated number of faces $\rho$ contained in the image, while the latter is allocated to the other areas that may consume the privacy budget through dichotomization. Unlike the Laplacian (LAP) algorithm, the noise error of the PPSA algorithm will not change with the image size, for the privacy protection is limited to the face area. The results show that the PPSA algorithm meets the requirements of $\varepsilon$-Differential privacy, and image classification is realized by using different image privacy protection algorithms in different human face databases. The verification results show that the accuracy of the PPSA algorithm is improved by at least 16.1%, the recall rate is improved by at least 2.3%, and $F_1$-score is improved by at least 15.2%.

1. Introduction

In the information age, the importance of data becomes more and more prominent. The value of data is highlighted in various fields in our society. Data-based services are widely applied in different industries. In particular, the application of image data has been developing rapidly. Thanks to the fast development of information technology and multimedia technology, it is now easier to acquire and share digital face images. The users can publish their own photos on social network platforms and other channels. Statistics show that more than 3.2 billion face images are shared by users of major social network platforms around the world. These digital images usually contain a wealth of personally sensitive information. If the information is collected and analyzed by attackers, unmeasurable losses will occur, in addition to the leak of personal privacy.

The privacy protection of image data often relies on techniques like $k$-anonymity, access control, and privacy encryption. Fung et al. [1] and Xiao and Tao [2] proposed the $k$-same method based on the anonymization mechanism. The method anonymizes each published gray image, reducing the probability of attackers deriving user identity from the published image to less than $1/k$. Li et al. [3] applied access control to restrict user access to social network images. The transfer and visitor volumes of these images are
reduced to protect privacy. This approach, which protects images via access settings, is not a fundamental privacy protection method for images. If an attacker has a certain background knowledge, he/she may bypass the access control and acquire the user images and privacy information. To prevent the communication process from being eavesdropped, Terrovitis et al. [4] implemented homomorphic encryption on gray images. However, every data encryption technique makes some assumptions and designs the corresponding encryption algorithm based on these assumptions. However new it is, the encryption algorithm will soon be breached by attackers. Focusing on anonymized images on Facebook, Sweeney, and Anonymity [5] revealed that attackers can deduce the social security number (SSN) of the person in each anonymized image, based on the unique Friendster feature of Facebook, and identify the sensitive information (e.g., disease and address) from the SSN.

In 2006, Dwork [6] proposed differential probability, which disturbs sensitive data by adding noise to the output. Differential privacy can hide the impact of a single record: whether a record is in the dataset or not, the output probability of the same result will not change significantly. In this way, attackers are hampered from further reasoning. Compared with other privacy protection techniques, the differential probability is superior in that it makes no assumption about the background knowledge of attackers. Dwork further investigated differential privacy in a series of theses [7–11] and proposed its implementation mechanism [12, 13]. McSherry [14] pointed out that the differential privacy algorithm for complex privacy problems needs to satisfy two composition properties: sequence combination and parallel combination. In recent years, differential privacy has been mainly applied to the field of data publication. The differential privacy protection for data publication mainly attempts to ensure the accuracy of published data or query results while satisfying differential privacy. The relevant research focuses on adjusting the publication mechanism and algorithm complexity. The primary research method is a quantitative analysis based on calculation and learning theories.

By the realization environment, the differential privacy protection for data publication can be divided into interactive data publication and noninteractive data publication [12]. On interactive data publication, Roth and Roughgarden [15] presented a median algorithm that can respond to more queries. Hardt and Rothblum [16] developed a pulse wave modulation (PMW) mechanism capable of increasing the number of queries. Gupta et al. [17] proposed a universal iterative dataset generation mechanism. Fan and Xiong [18] designed a novel approach with filtering and adaptive sampling for releasing time series under differential privacy (FAST). Kellaris et al. [19] put forward a flow data publication algorithm with no limit on the number of publications. On noninteractive data publication, Xiao et al. [20] came up with the Privelet algorithm. Xu et al. [21] created the noise first and structure first algorithms. Li et al. [22] proposed the matrix mechanism. Li et al. [23] put forward a data- and workload-aware algorithm (DAWA) algorithm.

Owing to the complexity of image data, the current research is still in the exploratory stage concerning the differential privacy protection for sensitive information in images.

Images are often represented as matrices in the real number field. Any pixel in an image can be mapped to a value at the corresponding location in a two-dimensional (2D) matrix. To satisfy $\varepsilon$-differential privacy, the most direct approach is to add a Laplace noise to all the values in the matrix. However, the disturbed image may be over distorted and become useless. Fourier transform and wavelet transform are common ways to compress images. Zhang et al. [24] proposed an image compression method based on discrete Fourier transform (DFT), which adds a unique Laplace noise to each target image. Despite suppressing the noise error, their approach introduces the reconstruction error to image compression. Considering the uncorrelation between the values in the image matrix, Liu et al. [25] converted the image gray matrix into a one-dimensional (1D) ordered data flow and modeled the data flow with the sliding window model. To protect the privacy of images, the privacy budget was allocated dynamically based on the similarity between the data of adjacent sliding windows. This highly available strategy is restricted to the 1D space. Liu et al. [26] utilized regional growth to expand the comparison between adjacent subgraphs to the 2D space and further optimized the privacy protection for face images.

The noises of the above methods cover the entire image. For face images, the sensitive information of the face only concentrates in the face area and even in some specific areas. The exposure of other nonsensitive areas will not lead to the leak of privacy. Therefore, this paper proposes a local privacy protection method for the sensitive areas in multiface images.

2. Background

2.1. Differential Privacy

Definition 1. Adjacent datasets of the face image

For a given image $X$, the gray matrix $X_{mn}$ can be obtained by normalizing the image. Then, there exists $X|X_{mn} = \begin{bmatrix} x_{11}, x_{12}, \ldots, x_{1n} \\ \vdots \ddots \vdots \\ x_{m1}, x_{m2}, \ldots, x_{mn} \end{bmatrix}$, where $x_{ij}$ in matrix $X_{mn}$ represents the gray of the corresponding element. If there exists an $X'$ with only one element difference from $X$, $|X - X'| = x_{ij} (1 \leq i \leq m, 1 \leq j \leq n)$, 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 $\text{Range}(M)$, the algorithm can provide $\varepsilon$-differential privacy, if its arbitrary outputs on two adjacent gray images $X$ and $X'$ satisfy the following:

$$\Pr[M(X) \in S] \leq \exp(\varepsilon) \times \Pr[M(X') \in S],$$

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

**Definition 3.** Global sensitivity

Let \( Q \) be a random query function meeting \( Q: D \rightarrow \mathbb{R}^d \). Then, the global sensitivity of \( Q \) can be expressed as follows:

\[
\Delta Q_{\text{GS}} = \max_{X,X'} \| Q(D) - Q(D') \|_p.
\]

**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 satisfies \( \varepsilon \)-differential privacy:

\[
M(D) = Q(D) + < \text{Lap} p_1(\frac{\Delta Q_{\text{GS}}}{\varepsilon}), \ldots, \text{Lap}_d(\frac{\Delta Q_{\text{GS}}}{\varepsilon}) >.
\]

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 \( f(x|\mu, b) = 1/2b e^{-|x-\mu|/b} \).

**Property 1.** Differential privacy-serial combination property

For a given dataset \( X \) and a set of differential privacy algorithms \( M_1(X), M_2(X), \ldots, M_m(X) \) related to \( X \), if the algorithm \( M_i(D) \) satisfies \( \varepsilon_i \)-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.

2.2. Multitask Cascaded Convolutional Network (MTCNN).

With the development of CNN [27], Sun et al. [28] suggested that the CNN can be applied to location face landmarks by virtue of its strong feature extraction ability. A three-layer deep CNN (DCNN) was designed to solve a thorny problem in landmark extraction: the inability to obtain the global optimal solution, due to the improper setting of initial values. Zhang et al. introduced multitask learning to face landmark localization and proposed a task-activated DCNN for face landmark localization (TCDCN). The TCDCN, a multitask learning model of four subtasks, has a smaller time complexity than traditional CNN [29]. Two years later, Zhang et al. developed the MTCNN, which effectively integrates face area detection with face landmark localization. The three cascading neural networks in MTCNN are responsible for face classification, bounding box regression, and key point localization, respectively [30]. Wu et al. [31] presented a tweaked CNN (TCNN), which relies on a mixed Gaussian model to cluster the features on different layers, and concluded that a deeper network layer can more accurately mirror face landmarks. In addition, many other methods have been adopted to locate face landmarks, namely, principal component analysis (PCA) [32], support vector machine [33], Bayesian probabilistic network (BPN) [34], dynamic link architecture (DLA) [35], and Gabor wavelet network (GWN) [36].

The MTCNN is a multitask parallel face recognition method based on deep learning. This method has been widely recognized in the industry because it operates rapidly, and its accuracy meets the general requirements of face detection. The multitask parallel capacity of the MTCNN algorithm manifests as the simultaneous detection of multiple faces (detecting whether an image contains faces and finding the face locations in the image, as shown in Figure 1) and the localization of face landmarks (locating the five landmarks, namely, two eyes, nose tip, and mouth corners, as shown in Figure 2).

The MTCNN algorithm can be realized as a three-stage cascaded CNN. The first stage is called a proposal network (P-Net), which mainly obtains the candidate windows for the face area and the regression vector of the bounding box, carries out regression with the bounding box, calibrates the candidate windows, and merges highly overlapped candidate boxes through nonmaximum suppression (NMS). The structure of the P-Net is shown in Figure 3.

The second stage is called a refine network (R-Net), which eliminates the false positive areas through bounding box regression and NMS. That is, lots of nonface windows are denied by a more complex CNN in order to refine the face window. The structure of the R-Net is shown in Figure 4.

The third stage is called an output network (O-Net). With one more convolutional layer than the R-Net, the O-Net outputs more refined results. The functions of the O-Net are the same as those of the R-Net. But the O-Net carries out more supervision of the face area and output five landmarks. The structure of the O-Net is shown in Figure 5.

Each stage of the MTCNN is a multitask network, responsible for handling tasks like face/ nonface judgement, face box regression, and landmark localization. The face/nonface judgement adopts the cross-entropy loss function:

\[
L_i^{\text{det}} = -(y_i^{\text{det}} \log(p_i)) + (1 - y_i^{\text{det}})(1 - \log(p_i)),
\]

where \( p_i \) is the probability for the image to contain faces; \( y_i^{\text{det}} \in \{0, 1\} \) is the true background label.

The regression loss of face box can be calculated by Euclidean distance:

\[
L_i^{\text{box}} = \| \tilde{y}_i^{\text{box}} - y_i^{\text{box}} \|_2,
\]

where \( \tilde{y} \) is the background coordinates predicted by the network; \( y \) is the true background coordinates; \( y_i^{\text{box}} \in \mathbb{R}^4 \) (\( \mathbb{R}^4 \) is a 4-tuple composed of \( x \) of the upper left corner, \( y \) of the upper left corner, length, and width).

Landmark localization is calculated similarly as face box regression. The Euclidean distance between the predicted landmark location and the true landmark location is calculated and minimized:

\[
L_i^{\text{landmark}} = \| \tilde{y}_i^{\text{landmark}} - y_i^{\text{landmark}} \|_2,
\]

where \( y_i^{\text{landmark}} \in \mathbb{R}^{10} \) is a 10-tuple, for five landmarks are being localized.
Figure 1: Face detection.

Figure 2: Landmark localization.

Figure 3: Structure of P-net.

Figure 4: Structure of R-net.
The simplest function of the training process can be expressed as follows:

$$\min \sum_{i=1}^{N} \sum_{j (\text{det, box, landmark})} \alpha_i \beta_j L_{ij}^i,$$  \hspace{1cm} (7)$$

where $N$ is the number of training samples; $\alpha_i$ is the importance of a task; $\beta_j^i \in [0, 1]$ is a sample label. In the P-Net and R-Net, $\alpha_{\text{det}} = 1$, $\alpha_{\text{box}} = 0.5$, and $\alpha_{\text{landmark}} = 0.5$. In the O-Net, $\alpha_{\text{det}} = 1$, $\alpha_{\text{box}} = 0.5$, and $\alpha_{\text{landmark}} = 1$.

### 3. Methodology

#### 3.1. Laplacian (LAP) Algorithm

This paper proposes the Laplacian algorithm based on the Laplace mechanism. In the LAP algorithm, every element $x_{ij}$ in the gray matrix $X_{\text{gray}}$ is regarded as an independent individual. This division method is the basis for applying the interactive mechanism to privacy protection. Each $x_{ij}$ (1 $\leq i \leq m$, 1 $\leq j \leq n$) consumes a privacy budget of $\epsilon/(m \times n)$. The LAP algorithm can be realized in the steps in Algorithm 1.

Compared with the original image $X$, the protected image $X'$ contains an additive noise of $2m \times n \times (\Delta Q \times m \times n/\epsilon)^2$. Admittedly, the LAP algorithm satisfies $\epsilon$-differential privacy. However, the noise results show that the LAP algorithm will have a huge error if it is applied to protect the privacy of an excessively large image. If so, the noisy image will be of low availability.

#### 3.2. Fusion Similarity Measurement Mechanism (FSMM)

This paper divides a face image $X_{\text{gray}}$ into multiple non-intersecting subgraphs $T_{ij}$ (1 $\leq i \leq I$, 1 $\leq j \leq J$), each of which contains multiple pixels:

$$X_{\text{gray}} = \begin{bmatrix} x_{11}, x_{12}, \ldots, x_{1m} \\ \vdots \vdots \vdots \vdots \\ x_{m1}, x_{m1}, \ldots, x_{mn} \end{bmatrix} = \begin{bmatrix} T_{11}, T_{12}, \ldots, T_{1j} \\ \vdots \vdots \vdots \vdots \\ T_{i1}, T_{i2}, \ldots, T_{ij} \end{bmatrix}. \hspace{1cm} (8)$$

Through the division, the nonintersecting subgraphs $T_{ij}$ carry as much information of the original image as possible.

The key to regional growth is to determine the criterion. Due to the natural complexity of face images, the criterion should be designed by judging the various features of the original image. The traditional criterion for regional growth only focuses on the difference between gray values. The gray value of a single pixel cannot provide rich information about the image. By contrast, subgraphs as the basic units of regional growth can retain the luminance, contrast, structure, color, texture, and spatial distribution features of the image. This paper presents a new regional growth criterion called FSMM, which further improves the accuracy of area merging with subgraphs as the basic units. The FSMM can be realized in the steps in Algorithm 2.

As its name suggests, the new regional growth criterion relies on the FSMM, a brand-new similarity measure between images. In rows 3-4 of the algorithm, parameter $\sigma$, a very small positive number, appears to ensure the denominator is nonzero. In rows 7-9, $\mu_X$ and $\mu_Y$ are, respectively, the mean values of images $X$ and $Y$, reflecting the luminance features of the images; $\sigma_X$ and $\sigma_Y$ are, respectively, the variances of images $X$ and $Y$, reflecting the contrast features; $C_1, C_2$ and $C_3$ are very small positive numbers that ensure the denominator is nonzero. In Column 10, $\alpha, \beta$, and $\gamma$ are parameters adjusting the proportion of different eigenvalues: if $\alpha = \beta = \gamma = 1$, then $C_1 = (K_1 L)^2, C_2 = (K_2 L)^2$, and $C_3 = C_2/2$, with $K_1 \ll 1, K_2 \ll 1$, and $L$ being the dynamic range of the image. The final calculation method for FSMM $(X, Y)$ can be expressed as follows:

$$\text{FSMM}(X, Y) = \frac{\phi \times (d(X, Y) / \rho(X, Y) + \sigma) \times (2\mu_X \mu_Y + C_1) \times (2\sigma_X \sigma_Y + C_2)}{(\mu_X^2 + \mu_Y^2 + C_1) \times (\sigma_X^2 + \sigma_Y^2 + C_2)} \hspace{1cm} (9)$$
Input: original image $X$, privacy budget $\epsilon$, parameters $m$ and $n$, subgraph similarity expectation $Th$
Output: Image $X'$ satisfying differential privacy
(1) Read the original image $X$, convert the image into gray matrix and store it in matrix $X_{m \times n}$
(2) for $i = 1$ to $m$
(3) for $j = 1$ to $n$
(4) $X'_{(i,j)} = X_{(i,j)} + \text{lap}(\Delta Q \times m \times n/\epsilon)$
(5) end for
(6) end for
(7) Output privacy protected picture $X'$

Algorithm 1: LAP.

Input: image $X$, image $Y$
Output: graph similarity $\theta$
(1) Convert the image to a grayscale image, $X' = \begin{bmatrix} X_1 & \cdots & \cdots \\ \cdots & \ddots & \cdots \\ \cdots & \cdots & X_c \end{bmatrix}$, $Y' = \begin{bmatrix} Y_1 & \cdots & \cdots \\ \cdots & \ddots & \cdots \\ \cdots & \cdots & Y_c \end{bmatrix}$
(2) Calculate the Euclidean distance between $X'$ and $Y'$, $\theta = \sqrt{\sum_{i=1}^{m}(X_i - Y_i)^2}$
(3) Calculate the Jaccard similarity index between $X'$ and $Y'$, $\rho(X', Y') = |X' \cap Y'|/|X' \cup Y'|$
(4) $\theta = \theta / \rho(X', Y') + \sigma$
(5) Combined with Hamming distance and perceptual hash algorithm $\phi$, $\phi = \begin{cases} 0.078 \times c \leq \mu(X', Y') \leq 0.156 \times c \\ (1/1g) \times (m \times n - \mu(X', Y')), \mu(X', Y') > 0.156 \times c \end{cases}$
(6) $\theta = \phi \times \theta$
(7) Calculate the luminance relationship between $X'$ and $Y'$, $l(X', Y') = (2u_X u_Y + C_3)/(u_X^2 + u_Y^2 + C_3)$
(8) Calculate the contrast ratio relationship between $X'$ and $Y'$, $c(X', Y') = (2u_X^2 \sigma_Y^2 + C_3)/(\sigma_X^2 + \sigma_Y^2 + C_3)$
(9) Calculate the structural relationship between $X'$ and $Y'$, $s(X', Y') = (\sigma_X^2 \sigma_Y^2 + C_3)/(\sigma_X^2 + \sigma_Y^2 + C_3)$
(10) $\text{SSIM}(X', Y') = [l(X', Y')]^{1/2} \cdot [c(X', Y')]^{1/2} \cdot [s(X', Y')]^{1/2}$
(11) $\theta = \text{SSIM}(X', Y') \times \theta$
(12) Output $\theta$

Algorithm 2: FSMM.

3.3. Privacy Protection for Sensitive Areas (PPSA) Algorithm.
To achieve privacy protection of sensitive information (face area) in multiface images and reduce noise impact on the privacy protected images, this paper proposes a local privacy protection method for face images, which combines face recognition, regional growth, and differential privacy. The proposed algorithm is called the PPSA.

Inspired by goal-driven reasoning (reverse reasoning), the PPSA was designed with goal as the starting point. The privacy protection for face images aims to prevent attackers from recognizing the target persons through face detection. Therefore, the first step of PPSA design is to understand how attackers identify faces. Next, a privacy protection method should be provided for face images, targeting the face identification technique of attackers. In this way, the designed privacy protection algorithm can effectively curb privacy leak and mitigate the effects of noise on the availability of the privacy protected image.

The design of the PPSA needs to solve three key problems: (1) locate the sensitive area in the multiface image and recognize the landmarks in the area; (2) formulate the criterion for regional growth; (3) allocate the privacy budget reasonably to areas requiring different levels of protection.

The PPSA algorithm is designed to protect the sensitive information (face information) in multiface images in batches. Different from single-face privacy protection, multiface privacy protection needs to deal with an unknown number of faces. Thus, the allocation of the privacy budget is crucial to PPSA design. The privacy protection should be tailored to areas with different protection requirements. In our scheme, the total privacy budget $\epsilon$ is divided into two parts: $\epsilon_1$ and $\epsilon_2$. The $\epsilon_1$ is evenly allocated to each seed $Se_{n}$ (any subgraph containing a landmark is a seed), according to the estimated number of faces $\rho$ contained in the image. During regional growth, $Se'$ (noisy seed) replaces the $T_{(i,j)}$ (adjacent subgraph) meeting the growth criterion, thereby protecting the privacy of $T_{(i,j)}$. Note that the privacy protection of $T_{(i,j)}$ does not consume any privacy budget. Meanwhile, the $\epsilon_2$ is allocated through dichotomization. This part of the privacy budget is consumed by the $Se_i$, outside the estimated number of faces $\rho$, the $T_{(i,j)}$ failing to meet the growth criterion, and the $T_{(i,j)}$ belonging to the sensitive area but not involved in regional growth. The PPSA algorithm can be realized in the steps in Algorithm 3.

The image preprocessing, privacy budget allocation, and regional growth are described in rows 1–6, rows 7–9, and rows 10–29, respectively. In row 30, $\hat{T}_{(i,j)}$ is the addition of Laplace noise to an unprotected subgraph in $S'$. In row 18, the PPSA limits the range of regional growth: if a $T_{(i,j)}$ meeting the growth criterion but lying beyond the coverage of the current $S_{i}$ or the growth range of the current $Se_{i}$, it will not be able to complete this area merging; the growth
Input: Original image X, privacy budget ε, estimated number of faces ρ, preset parameters β, estimate the number of faces in the image ω, subgraph similarity expectation Th
Output: Image X’ satisfying differential privacy.

(1) Read the original image X, convert the image into gray matrix and store it in matrix X_men.
(2) MTCNN is used to extract the face region S = (S_1, S_2, . . . , S_ω) and face feature point K_ω = (K_{1ω}, K_{2ω}, . . . , K_{mω}) in face image X
(3) Set the gray matrix X according to the preset parameters β. It is divided into subgraph sets T_(i,j) with the same structure
(4) If K_ω belongs to a subgraph T_(i,j), set T_(i,j) as Seed_n
(5) Find all backup Seed_N = (Seed_1, Seed_2, . . . , Seed_m)
(6) Adjust the face image area S as S’ according to the sub image size
(7) ε = ε_1 + ε_2
(8) T_2 = ε_1
(9) ε_2 = 2/2
(10) Seed = 0, Seed is used to record whether this seed has been used during regional growth
(11) Create a linked list to record the status of the current seed region merging process
(12) while (Seed ≤ 5 × ω)
(13) if (T_2 > 0)
(14) if (list_head == NULL)
(15) Pick unmarked Seed_n adds noise to it as Seed_n’, Seed_n’ = Seed_n + lap(ΔQ × 5 × ρ/ε_1)
(16) ε_1 = ε_1 - ε_2/5ρ
(17) Attempt to merge adjacent T_(i,j) start regional growth
(18) if (T_(i,j) ∈ S’ Euclidean_Distance (T_(i,j), Seed_n) ≥ Min (Euclidean_Distance (Seed_n, Seed_n)/2)
(19) brake
(20) else
(21) if (FSMM (Seed_n’, T_(i,j) ≤ Th)
(22) T_(i,j) = Seed_n’,
(23) else
(24) T_(i,j) = T_(i,j) + lap (ΔQ/ε_2)
(25) ε_2 = ε_2/2
(26) Seed_n set to marked, Seed = Seed + 1
(27) Release all data in the list, and the consolidation of this subregion is completed
(28) else if (ε_2 = 0)
(29) Execute line 15–line 28, line 16 ΔQ × 5 × ρ/ε_1 adjusted to ΔQ/ε_2, line 17 is adjusted to ε_2/2 = ε_2/2
(30) T_(i,j) = T_(i,j) + lap (ΔQ/ε_2)

Algorithm 3: PPSA.

Q.E.D.

**Theorem 2.** The PPSA consumes a privacy budget smaller than ε and satisfies ε-differential privacy.

Proof. In the PPSA, the total privacy budget ε is divided into two parts. Among them, ε_1 is used to add noise to seeds. If ω < ρ, there exists ε_1 > 0 after the end of the algorithm; if ω ≥ ρ, there exists ε_1 = 0 after the end of the algorithm. Meanwhile, ε_2 is allocated by dichotomy:

\[
\epsilon_2 \times \left(\frac{1}{2^n} + \frac{1}{2^n} + \frac{1}{2^n} + \cdots + \frac{1}{2^n}\right) < \epsilon_2 (N \rightarrow + \infty).
\]

Thus, there exists the following:

\[
\epsilon - \left(\frac{\epsilon_1}{2^n} + \frac{\epsilon_2}{2^n}\right) > 0.
\]

Therefore, the PPSA consumes a privacy budget smaller than ε. According to Property 1, the PPSA satisfies ε-differential privacy.
\[
\tilde{T}'_{(i,j),\text{LAP}} = \tilde{T}_{(i,j)} + \text{lap}\left(\frac{\Delta Q \times I \times J}{\varepsilon}\right)
\]
\[
\tilde{T}'_{(i,j),\text{PPSA}} = \tilde{T}_{(i,j)} + \text{lap}\left(\frac{\Delta Q \times 2^M}{\varepsilon_2}\right).
\]
(14)

If \(2^M \leq I \times J\), there exists Error (PPSA (\(P_{(i,j)}\))) < Error (LAP (\(P_{(i,j)}\))).

If \(2^M > I \times J\), there exists Error (PPSA (\(P_{(i,j)}\))) ≈ Error (LAP (\(P_{(i,j)}\))).

The validity of Error (PPSA (\(P_{(i,j)}\))) \approx Error (LAP (\(P_{(i,j)}\))) depends on the uniqueness of the gray image. In the gray image matrix, the gray value falls between 0 and 255. Under the Laplace mechanism, the smaller the \(\varepsilon\), the greater the noise. For any other type of data, the noise effect will be huge, if \(\varepsilon\) is sufficiently small. For gray value, however, the fluctuation induced by noise is limited. Take \(\tilde{T}'_{(i,j),\text{PPSA}} = \tilde{T}_{(i,j)} + \text{lap}\left(\Delta Q \times 2^M/\varepsilon_2\right)\) for example. When \(\text{lap}\left(\Delta Q \times 2^M/\varepsilon_2\right) \rightarrow +\infty\) holds, \(\tilde{T}'_{(i,j),\text{PPSA}}\) value in the gray matrix will become 255. When \(\text{lap}\left(\Delta Q \times 2^M/\varepsilon_2\right) \rightarrow -\infty\), that value will change to 0. Thus, Error (PPSA (\(P_{(i,j)}\))) = Error (LAP (\(P_{(i,j)}\))) holds, as long as \(I \times J\) is sufficiently great.

(3) For \(\text{Seed}_{n}\):

There are two possible scenarios: \(\omega \leq \rho\) and \(\omega > \rho\).

When \(\omega \leq \rho\), \(\text{Seed}'_{n,\text{DPLP}}\) only consumes \(\varepsilon_1\). Then, there exists the following:

\[
\text{Error}(\text{PPSA}(T_{(i,j)})) = T'_{(i,j),\text{PPSA}} = \text{Seed}_n + \text{lap}\left(\frac{\Delta Q \times 5 \times \rho}{\varepsilon_1}\right) - T_{(i,j)}
\]

\[
\text{Error}(\text{LAP}(T_{(i,j)})) = T'_{(i,j),\text{LAP}} = \text{lap}\left(\frac{\Delta Q \times I \times J}{\varepsilon}\right).
\]
(17)

Subtracting the above two formulas, we have the following:

\[
\text{Error}(\text{PPSA}(T_{(i,j)})) - \text{Error}(\text{LAP}(T_{(i,j)}))
\]
\[
= \text{Seed}_n + \text{lap}\left(\frac{\Delta Q \times 5 \times \rho}{\varepsilon_1}\right) - T_{(i,j)} - \text{lap}\left(\frac{\Delta Q \times I \times J}{\varepsilon}\right)
\]
\[
= \left(\text{Seed}_n + \text{lap}\left(\frac{\Delta Q \times 5 \times \rho}{\varepsilon_1}\right)\right) - \left(T_{(i,j)} + \text{lap}\left(\frac{\Delta Q \times I \times J}{\varepsilon}\right)\right)
\]
\[
= \text{Seed}_{n,\text{PPSA}} - T_{(i,j),\text{LAP}}.
\]
(18)

For a gray image, when the privacy budget is sufficiently small under the Laplace mechanism, Error (\(T_{(i,j),\text{LAP}}\)) is equivalent to Error (\(\text{Seed}_{n,\text{LAP}}\)). Thus, we have the following:

\[
\text{Seed}_{n,\text{PPSA}} = \text{Seed}_n + \text{lap}\left(\frac{\Delta Q \times 5 \times \rho}{\varepsilon_1}\right)
\]
(15)

If \(5 \times \rho \ll I \times J\), the numerical gap between \(\varepsilon_1\) and \(\varepsilon\) is so small as to be negligible. Subtracting the above two formulas, we have the following: \(\text{Error (PPSA} (\text{Seed}_n)) < \text{Error (LAP} (\text{Seed}_n))\).

When \(\omega > \rho\), the first \(\rho\) face areas consume \(\varepsilon_1\) until no budget in this part is left. In this case, there exists \(\text{Error (PPSA (Seed}_{n})) < \text{Error (LAP (Seed}_{n}))\). For the lack of space, the proof is omitted.

In the following \(\omega - \rho\) face images, \(\text{Seed}_{n,\text{DPLP}}\) needs to consume \(\varepsilon_2\). According to the proof of \(T_{(i,j)}\), there exists \(\text{Error (PPSA (Seed}_{n})) = \text{Error (LAP (Seed}_{n}))\) during the consumption of \(\varepsilon_2\). Thus, it can be deduced that, in any scenario, there exists the following:

\[
\text{Error (PPSA (Seed}_{n}))_{\varepsilon_1} + \text{Error (PPSA (Seed}_{n}))_{\varepsilon_2}
\]
\[
< \text{Error (LAP (Seed}_{n}))_{\varepsilon_1}
\]
(16)

(4) For \(T_{(i,j)}\):

There are also two possible scenarios: \(\omega \leq \rho\) and \(\omega > \rho\). When \(\omega \leq \rho\), the proof is as follows:

\[
\text{Seed}_{n,\text{PPSA}} - T'_{(i,j),\text{LAP}}
\]
\[
= \text{Seed}_{n,\text{PPSA}} - \text{Seed}_{n,\text{LAP}}
\]
\[
= \text{Seed}_n + \text{lap}\left(\frac{\Delta Q \times 5 \times \rho}{\varepsilon_1}\right) - \text{Seed}_n - \text{lap}\left(\frac{\Delta Q \times I \times J}{\varepsilon}\right)
\]
\[
= \text{lap}\left(\frac{\Delta Q \times 5 \times \rho}{\varepsilon_1}\right) - \text{lap}\left(\frac{\Delta Q \times I \times J}{\varepsilon}\right) < 0.
\]
(19)

Thus, \(\text{Error (PPSA} (T_{(i,j)})) < \text{Error (LAP} (T_{(i,j)}))\).

Similarly, when \(\omega > \rho\), there exists \(\text{Error (PPSA} (T_{(i,j)})) \approx \text{Error (LAP} (T_{(i,j)}))\). For the lack of space, the proof is omitted.

Q.E.D.
4. Experiments and Results Analysis

4.1. Experiments. To demonstrate its feasibility, the PPSA was tested on a 750 * 1020 image containing 8 faces from the WIDER FACE dataset. The original image was divided into 7,650 subgraphs of the size 10 * 10. During the implementation of the PPSA, the original image (Figure 6) was first transformed into a gray image (Figure 7). Then, the MTCNN algorithm was called to recognize the face area $S$ and landmark $K_S$ (Figure 8). To meet the requirements of the PPSA on regional growth, the face image was divided into multiple equal size $T_{(i,j)}$ (Figure 9), using the preset parameters. The subgraph area (Figure 10) overlapping the landmark location was taken as the Seed (Figure 11) of regional growth. However, the $S$ detected by the MTCNN is not compatible with the size $f T_{(i,j)}$ (Figure 12). To solve the problem, $S$ was resized to $S'$ (Figure 13), according to the size of $T_{(i,j)}$.

Figure 14 provides the results of the LPA. Figure 15 presents the PPSA results under the same privacy budget.
Observations show that the PPSA operates within the face area while the LAP acts across the image. Besides, within the scope of the regional growth algorithm, the PPSA retained the recognizability of landmarks. Although not excluding the possibility of privacy leaks, this strikes a balance between the privacy and availability of the privacy protected multiface image. In addition, the relationship between the number of faces \( \rho \) estimated by the PPSA and the true number of faces \( \omega \) determines how much privacy budget can be obtained by a seed. The allocation of the privacy budget will directly affect the error of the protected image. Figures 16 and 17 present the local results at different relationships between \( \rho \) and \( \omega \).

4.2. Results Analysis. To verify the feasibility of our algorithm, multiple face images were collected from WIDER FACE dataset, i-bug face dataset, and AFW dataset and subjected to experiments using TensorFlow + AlexNet CNN. The experiments were carried out in
the environment of Intel® Core i9-9900K CPU @ 3.60 GHz, 32G memory, GTX 21080TI GPU, and Windows 10. The privacy budget was set to 1, 2, 3, 4, and 5, in turn. The face recognition performance was measured by precision, recall, and F1-score. The experimental results are displayed in Figures 18–26.

The experimental results show that the PPSA achieved different results on images from different datasets. The difference arises from the varied image sizes of different datasets: 760 * 1000 in WIDER, 500 * 800 in i-bug, and 1000 * 1500 in AFW. In addition, the different background complexities between the face datasets also influence the operation effects of LAP, sliding window publication (SWP) algorithm, and region growing publication (RGP). By contrast, the operation of the PPSA is not affected by image size and background complexity.
Figure 18: WIDER and precision.

Figure 19: WIDER and recall.

Figure 20: IDER and F1-score.

Figure 21: 1-bug and precision.

Figure 22: 1-bug and recall.

Figure 23: 1-bug and F1-score.
From the experimental results, it can be learned that the operation results of the PPSA mainly depend on the number of estimated faces $\rho$ and $S'$. In the images from i-bug, there are relatively few faces in each image, but the face area takes up a large portion of the entire image. In the images from WIDER, there are many faces in each image, and the face area takes up a large portion of the entire image. In the images from the AFW, there are relatively few faces in each image, but the face area takes up a small portion of the entire image, for the images tend to be large. Therefore, it is expected that the PPSA’s privacy protection effect is positively correlated with $\omega$, and negatively with $S'$. The prediction is consistent with the experimental results.

5. Conclusions

To protect the privacy of multiface images, this paper combines face detection, regional growth, with the Laplace mechanism of differential privacy to add noises to local sensitive areas in face images and realizes the privacy protection for the local sensitive areas in multiface images under an interactive framework. Compared with the LAP, SWP, and RGP, the proposed PPSA can effectively suppress the noise impact on the protected image and improve the availability of the privacy protected image. Moreover, the PPSA applies to images of various sizes, and its error does not increase with the image size.

Although the PPSA can effectively protect the face information in multiface images, the attackers may choose to attack the hair, clothes, and body of persons in the images. Besides, privacy leak may arise from the relationship between persons, and the correlation between each person and the background. The future work will further improve image privacy protection, trying to solve these potential risks.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors thank the project of the National Natural Science Foundation of China (No. 61672179), Natural Science Foundation of Heilongjiang Province (No.
References

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

[2] X. Xiao and Y. Tao, “Anatomy: simple and effective privacy preservation,” in Proceedings of the 32nd IntConf on Very Large Data Bases, pp. 139–150, ACM, Seoul, Korea, September, 2006.

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

[4] 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.

[5] L. Sweeney and K. Anonymity, “A model for protecting privacy,” International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 10, no. 5, pp. 557–570, 2002.

[6] C. Dwork, “Differential privacy,” in Proceedings Of The 33rd International Colloquium On Automata Languages And Programming, pp. 1–12, Berlin, 2006.

[7] 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, April, 2008.

[8] C. Dwork, “The differential privacy frontier (Extended Abstract),” Theory of Cryptography, in Proceedings of the 6th Theory of Cryptography Conference, pp. 496–502, San Francisco, CA, USA, March, 2009.

[9] 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, January, 2009.

[10] C. Dwork, “Differential privacy in new settings,” in Proceedings of the Symposium On Discrete Algorithms (SODA), January, 2010.

[11] C. Dwork, “The promise of differential privacy,” A Tutorial on Algorithmic Techniques,” in Proceedings of the 52nd Annual IEEE Symposium on Foundations of Computer Science, Palm Springs, CA, USA, October, 2011.

[12] 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, March, 2006.

[13] F. McSherry and K. Talwar, “Mechanism design via differential privacy,” in Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science, pp. 94–103, Providence, Rhode Island, USA, October, 2007.

[14] F. McSherry, “Privacy integrated queries: an extensible platform for privacy-preserving data analysis,” Communications of the ACM, vol. 53, no. 9, pp. 89–97.

[15] A. Roth and T. Roughgarden, “Interactive privacy via the median mechanism,” in STOC ’10: Proceedings of the forty-second ACM symposium on Theory of computing, pp. 765–774, New York, NY, USA, June 2010.

[16] M. Hardt and G. N. Rothblum, “A multiplicative weights mechanism for privacy-preserving data analysis,” in Proceedings of the Foundations Of Computer Science (FOCS), 2010.

[17] A. Gupta, A. Roth, and J. Ullman, “Iterative Constructions and Private Data Release,” in Proceedings of the Theory of Cryptography, pp. 339–356, Berlin Heidelberg, July, 2012.

[18] L. Fan and L. Xiong, “Real-time Aggregate Monitoring with Differential Privacy,” in Proceedings of the Acm International Conference on Information & Knowledge Management, Glasgow Scotland, UK, October, 2012.

[19] G. Kellaris, S. Papadopoulos, X. Xiao, and D. Papadias, “Differentially private event sequences over infinite streams,” Proceedings of the Vldb Endowment, vol. 7, no. 12, pp. 1155–1166, 2014.

[20] X. Xiao, G. Wang, and J. Gehrke, “Differential private via wavelet transforms,” IEEE Transactions on Knowledge and Data Engineering, vol. 23, no. 8, pp. 1200–1214, 2014.

[21] J. Xu, Z. Zhang, X. Xiao, Y. Yang, G. Yu, and M. Winslett, “Differentially private histogram publication,” The VLDB Journal, vol. 22, no. 6, pp. 797–822, 2013.

[22] C. Li, G. Miklau, M. Hay, A. McGregor, and V. Rastogi, “The matrix mechanism: optimizing linear counting queries under differential privacy,” The VLDB Journal, vol. 24, no. 6, pp. 757–781, 2015.

[23] C. Li, M. Hay, G. Miklau, and Y. Wang, "A data- and workload-aware algorithm for range queries under differential privacy," Proceedings of the VLDB Endowment, vol. 7, no. 5, pp. 341–352, 2014.

[24] 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.

[25] C. Liu, J. Yang, W. Zhao, Y. Zhang, J. Li, and C. Mu, “Face image publication based on differential privacy,” Wireless Communications and Mobile Computing, vol. 2021, Article ID 6680701, 20 pages, 2021.

[26] C. Liu, J. Yang, W. Zhao et al., “Differential privacy protection of face images based on region growing,” Traitement du Signal, vol. 38, no. 5, pp. 1385–1401, 2021.

[27] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.

[28] Y. Sun, X. Wang, and X. Tang, “Deep convolutional network cascade for facial point detection,” in Proceedings of the Computer Vision and Pattern Recognition, pp. 3476–3483, Portland, OR, USA, June, 2013.

[29] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, “Facial landmark detection by deep multitask learning,” in Proceedings of the European Conference on Computer Vision, pp. 94–108, Pasadena, CA, UK, September, 2014.

[30] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, “Joint face detection and alignment using multitask cascaded convolutional networks,” IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499–1503, 2016.

[31] Y. Wu, T. Hassner, K. Kim, G. Medioni, and P. Natarajan, “Facial landmark detection with tweaked convolutional neural networks,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 12, pp. 3067–3074, 2017.

[32] N. Y. Ke and R. Sukthankar, “PCA-SIFT: a more distinctive representation for local image descriptors,” in Proceedings of the IEEE Computer Society Conference on Computer Vision & Pattern Recognition, Washington, DC, USA, July, 2004.

[33] C. Liu, J. Yang, and J. Q. Wu, “Web intrusion detection system combined with feature analysis and SVM optimization,” EURASIP Journal on Wireless Communications and Networking, vol. 2020, no. 1, pp. 1–11, 2020.
[34] W. Zhang, X. Li, Q. Song, and W. Lu, "A face detection method based on image processing and improved adaptive boosting algorithm," *Traitement du Signal*, vol. 37, no. 3, pp. 395–403.

[35] L. Wiskott, J.-M. Fellous, N. Kuiger, and C. Von Der Malsburg, "Face recognition by elastic bunch graph matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 775–779, 1997.

[36] N. Jiang and J. Li, "An improved semantic segmentation method for remote sensing images based on neural network," *Traitement du Signal*, vol. 37, no. 2, pp. 271–278, 2020.