sensors

Article

GAN-Based Differential Private Image Privacy Protection Framework for the Internet of Multimedia Things

Jinao Yu [1], Hanyu Xue [2], Bo Liu [2], Yu Wang [3,*], Shibing Zhu [1] and Ming Ding [4]

1 School of Space Information, Space Engineering University, Beijing 101416, China; yujinaosay@163.com (J.Y.); zhushibingsay@163.com (S.Z.)
2 School of Computer Science, University of Technology Sydney, Sydney, NSW 2007, Australia; Hanyu.Xue@student.uts.edu.au (H.X.); bo.liu@uts.edu.au (B.L.)
3 Institute of Artificial Intelligence and Blockchain, Guangzhou University, Guangzhou 510006, China
4 Data61, CSIRO, Sydney, NSW 2015, Australia; Ming.Ding@data61.csiro.au
* Correspondence: yuwang@gzhu.edu.cn

Abstract: With the development of the Internet of Multimedia Things (IoMT), an increasing amount of image data is collected by various multimedia devices, such as smartphones, cameras, and drones. This massive number of images are widely used in each field of IoMT, which presents substantial challenges for privacy preservation. In this paper, we propose a new image privacy protection framework in an effort to protect the sensitive personal information contained in images collected by IoMT devices. We aim to use deep neural network techniques to identify the privacy-sensitive content in images, and then protect it with the synthetic content generated by generative adversarial networks (GANs) with differential privacy (DP). Our experiment results show that the proposed framework can effectively protect users' privacy while maintaining image utility.

Keywords: Internet of Multimedia Things (IoMT); image privacy; object detection; deep learning; generative adversarial network; differential privacy

Citation: Yu, J.; Xue, H.; Liu, B.; Wang, Y.; Zhu, S.; Ding, M. GAN-Based Differential Private Image Privacy Protection Framework for the Internet of Multimedia Things. *Sensors* **2021**, *21*, 58. https://dx.doi.org/10.3390/s21010058

Received: 24 November 2020
Accepted: 19 December 2020
Published: 24 December 2020

1. Introduction

The recent advances in multimedia recording devices, such as phones, cameras, drones, and other types of sensors, have greatly facilitated the collection of multimedia data, especially in the form of images and videos. In such an era of IoMT, a massive number of images are widely used, not only by social network personal users but also by government and companies. Image data is the most representative type of data in an IoMT data collection and can contain sensitive information that might be used to uncover personal information. The relationship between IoMT sensors (phones, cameras, drones, monitoring cameras) and personal private information shows in Figure 1.

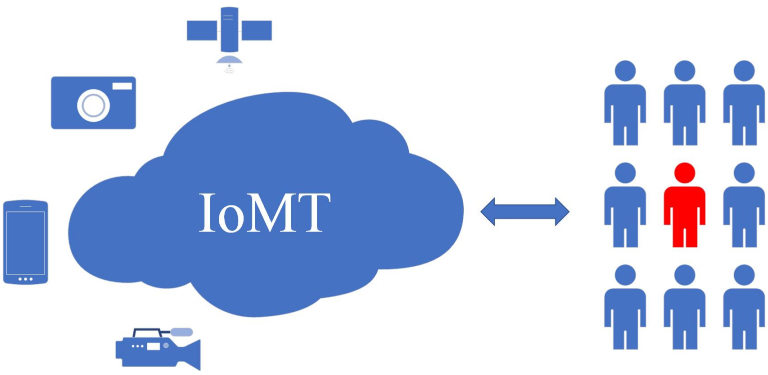

Figure 1. The IoMT collects sensitive private data through sensors (phones, cameras, drones, monitoring cameras) that might leak personal privacy.

Data mining attacks on images can easily cause privacy leakage, which can have serious consequences. The issue of privacy leakage has attracted the attention of the public in recent years and has aroused public concern about this issue. Moreover, privacy issues are no longer just personal concerns as many countries have launched privacy acts and laws. For example, the European General Data Protection Regulation (GDPR) took effect on 25 May 2018 [1]. Any violations of the regulation will trigger heavy fines and penalties. GDPR emphasizes the protection of "personal data", interpreting this as "any information relating to an identified or identifiable natural person ('data subject'); an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person" [2]. According to this definition, images include a variety of personal identifiers such as people's faces, text and license plates. Therefore, effective image privacy protection techniques are in urgent need. The research community has expended considerable effort on image privacy protection. The early works mostly focus on the access control of the data, i.e., privacy protection by safeguarding against unauthorized access. This can be achieved by setting the preferences of users [3,4] or tag controls [5,6]. However, these methods cannot be applied to scenarios where images are shared openly, but some sensitive information needs to be concealed. For example, in the "Google Street View" application, we have full access to photos showing the streets while people's faces and other personal identifiers have been obfuscated, e.g., by blurring. To achieve this, privacy protection methods need to detect, and then cover/remove/replace sensitive content in images. Several recent studies have explored this direction [7–16]. For example, Viola et al. [8] used a sliding window detector to identify and blur the license plates in Google Street View images. Yu et al. [11] used a deep multitask learning algorithm to detect privacy-sensitive objects and provide simple protection by blurring. Overall, most of the existing work undertakes personal data detection as the first step in privacy protection, relying on simple approaches such as blurring or pixelation. Consequently, the image utility suffers to a considerable extent. It not only makes the images look unnatural, but the person who looks at the image is aware that the obfuscated part is private. Moreover, such a protection mechanism is powerless against the emerging attacks based on advanced deep neural networks. For example, McPherson et al. [17] use artificial neural networks to recover hidden information from images protected by pixelation, blurring and P3. The method obtained good results on different data sets, MINIST 80%, CIFAR-10 75%, AT&T dataset 95%, FaceScurb 57%.

Moreover, the existing methods almost entirely focus on single object protection, such as face or text. However, most images that require privacy protection have multiple objects that need to be protected (For example, in street view images, human faces and license plates need to be protected at the same time).

Current methods are unable to find a way to quantify the tradeoff between image usability and privacy protection. To tackle this, we use DP to control the generation of de-identify objects in images to mitigate privacy threats.

To overcome these obstacles, we propose a three-stage framework for image privacy protection in this paper. The framework consists of three steps: (1) privacy-sensitive content detection and position extraction powered by a deep convolutional neural network: we use CNN networks to detect various objects in images and classify objects as either private or non-private; (2) projecting real private objects into latent space: we use generative adversarial networks (GANs) to project the real private objects of the images into latent space and obtain the corresponding latent vector ω^*. (3) private content generation controlled by DP (de-identification): we use Laplace noise into the latent vector ω^* and to generate the de-identification content. Finally, we replace the originally private objects with the synthetic ones to protect users' privacy.

In order to evaluate the performance of our proposed framework, we conduct extensive experiments on a real-world image data set collected by IoMT cameras, and investigate two types of personal identifier-related data: license plate and face. We choose these two types of objects as they represent the two most significant categories of personal identifiers in images.

In summary, the contributions of this paper are as follows:

- We propose an image privacy protection framework that can protect the privacy in IoMT images.
- We propose a GAN-based method to generate the replacement content for private objects in images.
- We use differential privacy methods to control image generation between image usability and privacy protection.

The remainder of the paper is organized as follows. In Section 2, we review the related work. In Section 3, we give the definition and foundation of the methods. In Section 4, we present our framework on multimedia privacy protection based on Mask-RCNN and synthetic content generation using GANs. In Section 5, we detail the experiment results of our framework for multiobject privacy protection (street view scenarios). In Section 6, we conclude the paper and outline future work.

2. Related Work

Privacy protection, in general, has been extensively studied in recent years. Of all the research in this area, differential privacy (DP) has attracted the most attention and has been applied to many different applications. Therefore, in this section, we review the most relevant research on image privacy and the related fundamental deep learning research, including: (1) image privacy issues and protection; (2) deep learning and object detection of images; (3) content generation; and (4) privacy protection.

2.1. Image Privacy Issues and Protection

Image privacy issues first attracted attention with the raipd development of social networks. The proliferation of social networks generated a massive number of photos flooding the Internet, some of which contain sensitive information. For example, Tang et al. [18] illustrated the problem of unpermitted photo sharing in social media and Pesce et al. [19] investigated the use of photo tags by third parties to compromise a user's privacy. The image privacy issue becomes more severe with the widespread use of facial recognition systems, as people worry that their faces might be used by organizations for profiling or social control.

To combat image privacy attacks, the previous mainstream method uses access control on sensitive content. Mannan et al. [3] use instant messaging (IM) networks to control personal web content sharing. Vyas et al. [4] use annotation data to predict the privacy preferences of users and control shared content. Wang et al. [5] studied privacy control on Facebook, and Squicciarini et al. [6] proposed collaborative privacy management to enable users to collaboratively control their photos. Similarly, to deal with the privacy issue in facial recognition systems, the current countermeasure is simply banning [20]. However, an access-control-based method has several limitations. It only gives a "yes" or "no" option for the use of images, whereas parts of the information in images need to be used by applications such as Google Street View, and as it cannot automate privacy protection based on the privacy information of the image itself, it requires human participation.

More recent image privacy research focuses on the inherent implicit information of the photos. Tonge et al. [10] explore learning models that can automatically classify the private or public parts in an image using deep neural networks. Yu et al. [11] create a new tool called "iPrivacy" which uses a deep learning algorithm to detect privacy-sensitive objects. Yu's work can detect the private parts of photos, but in the privacy protection step, they only use blur to protect privacy which does not look good. Uittenbogaard's work [12] goes one step further than blurring and sets a framework that automatically removes moving objects. However, there are two limitations, one is that it is only for moving objects and the other limitation is that it missed partial information of the image. Liu's work [13] proposes a novel Stealth algorithm, which prevents an automatic detector from detecting the objects in an image. However, humans can easily extract private information from an image.

Our framework is a further advancement compared with the aforementioned research. It can identify the private parts of photos at the pixel level. Then, it generates the target replacement content based on the private content, not just using a mosaic, blurring or removal to protect privacy. Our framework can protect private information from both humans and machines.

2.2. Deep-Learning-Based Image Object Detection and Segmentation

Object detection and semantic segmentation technologies have been advancing rapidly in recent years. In the beginning, Girshick et al. [21] used high-capacity convolutional neural networks (CNNs) for bottom-up region proposals, called R-CNN. This algorithm improves the mean average precision (mAP). In 2015, Hariharan et al. [22] defined the hypercolumn at a pixel as the vector of activations of all CNN units above that pixel to improve the results of the experiment. After this, of the majority of the research is based on the Fast R-CNN [23,24] and Fully Convolution Network (FCN) [25]. The disadvantage of Faster R-CNN is that it cannot deal with pixel-to-pixel alignment between the inputs and outputs of the network. To solve this problem, He et al. proposed a method called Mask R-CNN [26] that extends the Fast R-CNN by adding prediction segmentation masks on each region of interest (RoI) to get the results. As our goal is to find the private parts of images, we use Mask R-CNN to obtain the instance segmentation results that can be used as the basis for the follow-up privacy content detection and positioning. To obtain good results for our use case, we need to re-train the network using our image dataset which includes more privacy-sensitive content.

2.3. GAN-Based Content Generation

Preliminary ways to perform image privacy content protection include blurring, deletion, etc. In this paper, we replace content to protect privacy, i.e., generating content without any identifying information to replace the private content in the images. Traditional content generation methods such as [27–30] merely fill the pixels by matching and pasting based on the low-level features in the images. The effect is not very satisfactory as they often produce error content and the results obtained are also not good. In 2014, Goodfellow proposed a new framework called GAN [31] which can synthesize new content by training the models. Following the GAN-based method, the latest GAN-based generation content generation technology can generate very realistic content, such as faces, cats, dogs, even Airbnb rooms [32–35]. In our framework, we use StyleGAN [36] to generate the replacement content. StyleGAN can generate content which is not much different from the real image. The image content generated by StyleGAN does not exist in real life and this content can avoid copyright disputes. With the replacement of the generated content, the privacy of the images can be protected.

2.4. Privacy Protection

Privacy protection is an essential component for information system and management, such as network security [37,38], reputation management [39], blockchains [40] and the next generation of communication systems [41]. In traditional privacy protection technology, one of the most common methods is data encryption, which has high security. However, directly encrypting and decrypting large-scale data such as image sets will consume a lot of computing resources. Another privacy protection method is anonymity privacy protection technology. In 2002, Sweeney et al. proposed the k-Anonymity [42] method to protect privacy. Machanavajjhala proposed l-Diversity [43] to address the limitations of k-Anonymity, and Li et al. introduced t-Closeness [44]. However, with the development of attack technology, attackers can use data mining, machine learning, background knowledge attack, and big data analysis to obtain enough useful information on a user's privacy.

To solve this problem, Dwork [45] proposed the concept of differential privacy which has a solid mathematical theoretical foundation. Once differential privacy was proposed, it attracted attention in the field of privacy protection, and various privacy protection algorithms based on differential privacy have been proposed. In this paper, we propose a new image privacy protection method based on the differential privacy method combined with GANs. Taking advantage of the controllability of differential privacy, our method can protect the privacy of IoMT images with high controllability.

3. Preliminaries

In this section, we discuss image privacy protection, image utility, provide the basic knowledge and equations definition of our method.

3.1. Privacy Protection and Image Utility

In this part, we discuss image privacy protection and image utility. Firstly, the different levels of image privacy risk are shown in Figure 2. On the left are the images that do not contain any private information (such as a landscape photograph) and the risk of privacy leakage is zero. On the right are the images that contain private information which can be linked to specific individuals which violates individuals' privacy directly. Between the two extreme cases are images that contain private information but might not leak individuals privacy. Our goal is to propose a framework to reduce the risk of privacy leak from Level 3/4 back to Level 2 in Figure 2. This means that we can protect the private information in images so that they cannot be linked to any individual.

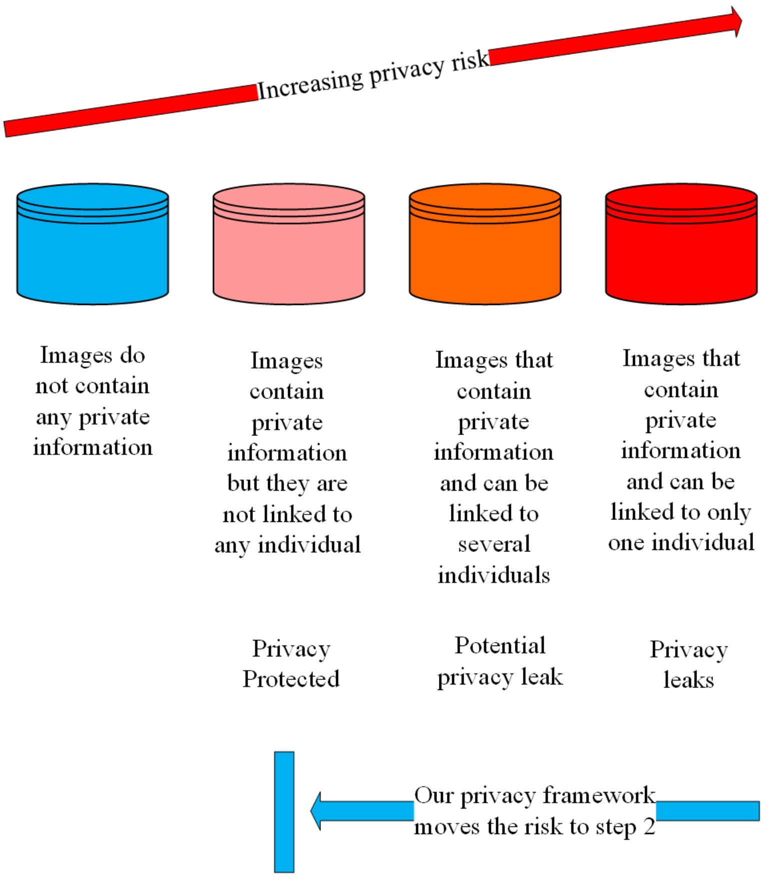

Figure 2. The four levels of image privacy risks.

However, the strength of privacy protection will affect the utility of images. The common methods such as mosaic and blur, might reduce the utility of the image while image processing. Greater privacy protection, results in lower utility of images, as shown in the example in Figure 3. Although the mosaic and blur methods protect privacy, it reduces the readability and usability of the images. It also makes image sharing pointless. In our image privacy protection framework, we have developed an effective way to compromise between privacy protection and image utility.

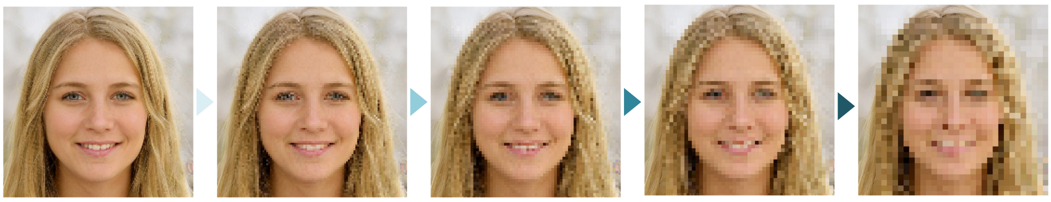

Figure 3. The privacy and utility.

3.2. Formulation of Image De-Identification

We now formally define the problem of image de-identification which enables us to define the problem we need to address and build the foundation for the following discussions.

Definition 1 (***Image***)**.** *An image is a matrix I of m columns, n rows and c channels. There are usually 3 channels in the common color space, such as RGB and YUV. Each cell in matrix I contains a coding which ranges from 0 to 255. Images should contain multiprivate objects such as face or text.*

Definition 2 (***Object sets***)**.** *An object set is a set of M object images contained in image matrix I: $O_i : i = 1, 2, ..., M$.*

Definition 3 (***Private object sets***)**.** *A private object set is a set of N objects images contained in image matrix I: $P_i : i = 1, 2, ..., N$ in which $P_i \in O_i$ and $N \leq M$.*

Definition 4 (***Private Object De-Identification Function***)**.** *Let $\mathbf{P}$ and $\mathbf{P}_d$ be a private object set and a de-identification object set.*

$$g : \boldsymbol{P} \rightarrow \boldsymbol{P}_d \tag{1}$$

where g is defined as the de-identification function for each P_i to remove their identity.

Definition 5 (***Image De-Identification***)**.** *Given image matrix I and de-identification function g, for each private object $P_i \in O_i$:*

$$I_d = g(I) \tag{2}$$

in which we can use the de-identification function to get an image matrix I_d with no private information.

3.3. Differential Privacy

Definition 6 (***Differential Privacy***)**.** *The formal definition of DP is given by Equation (3):*

$$Pr[K(D_1) \in S] \leq exp(\epsilon) \times Pr[K(D_2) \in S] \tag{3}$$

Definition 7 (***The Sensitivity of Differential Privacy***)**.** *The sensitivity of DP is defined in Equation (4), which determines how much perturbation is required in the DP mechanism.*

$$\Delta f = \max_{D_1, D_2} ||f(D_1) - f(D_2)||_1 \tag{4}$$

4. Image De-Identification Framework

In order to achieve the goal of image privacy protection, we propose an image de-identification framework which comprises three steps: (a) object detection and private object extraction; (b) de-identification content generation; and (c) content replacement and image privacy protection.

Figure 4 shows the diagram of the framework. The original image I contains private information such as a face or a car license plate. It is first input into a CNN to identify and extract the private objects in the image. Then we transform the extracted private objects into latent space and use differential privacy to control the de-identified content generation. Finally, we obtain a de-identified image I', i.e., an image without any sensitive information. In the following part of this section, we explain the framework in detail.

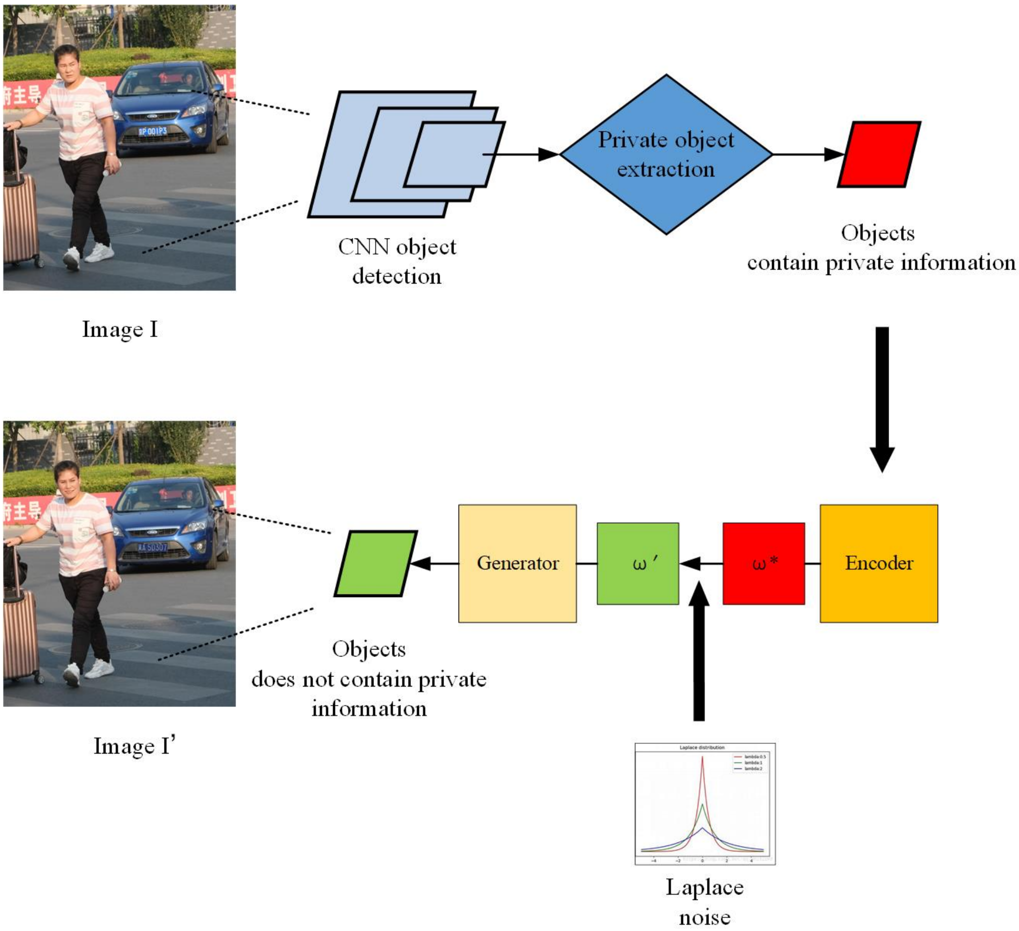

Figure 4. The diagram of the proposed image de-identification (DE-ID) framework.

4.1. Step-I: Object Detection and Private Object Extraction

To protect the privacy of an image, it is necessary to detect the sensitive privacy zones in the image. We use two steps to achieve this. First, all the objects in the image are detected, and then the private objects are extracted.

4.1.1. Object Detection

The state-of-the-art object detection algorithm Mask-RCNN is used to detect the objects in the image. The diagram of object detection is shown in Figure 5. Images contain private information are detected by object detection algorithm Mask-RCNN, and the Mask-RCNN can detect all objects and position of the Images.

For an image I, the ROI vector $\boldsymbol{X}_{roi}$ of each object O_i can be detected by $R(\cdot)$:

$$\boldsymbol{X}_{roi} = R(I) = (\boldsymbol{S}|\boldsymbol{E})$$

$$= \left(\begin{array}{cccc|cccc} x_1 & y_1 & w_1 & h_1 & E_{11} & E_{12} & \dots & E_{1m} \\ x_2 & y_2 & w_2 & h_2 & E_{21} & E_{22} & \dots & E_{2m} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & w_n & h_n & E_{n1} & E_{n2} & \dots & E_{nm} \end{array} \right) \tag{5}$$

where $\boldsymbol{S}_n = (x_n, y_n, w_n, h_n)$ is the position vector including the information of the upper left corner coordinate (x_i, y_i), width w_i and height h_i of object O_i. The probability of objects noted as $\boldsymbol{E}$, the $\boldsymbol{E}_i$ is the probability of Object O_i belonging to the mth class (there are m class objects in the image I).

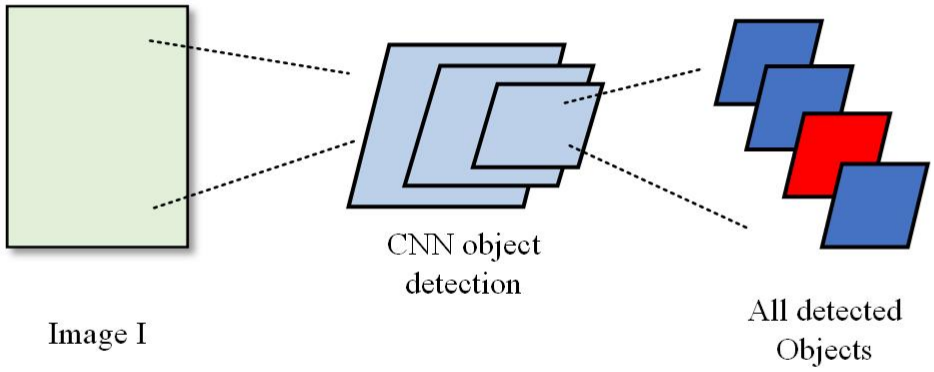

Figure 5. The diagram of object detection part in our image de-identification (DE-ID) framework.

In Equation (5), we choose the maximum probability c_i in each $\boldsymbol{E}_i$, so the output of the object detection is shown as follows:

$$\mathbf{O} = (\boldsymbol{S}|\boldsymbol{C}) = \left(\begin{array}{cccc|c} x_1 & y_1 & w_1 & h_1 & c_1 \\ x_2 & y_2 & w_2 & h_2 & c_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & w_n & h_n & c_n \end{array}\right) \tag{6}$$

where $\forall i \in (1, n)$:
$c_i = \begin{cases} \arg max(E_{ij}), 1 \leq j \leq m; \text{if } max(E_{ij}) > \delta \\ c_{bg}, \text{if } max(E_{ij}) \leq \delta \end{cases}.$

In Mask-RCNN, if the maximum probability is smaller than a threshold δ, this object will be treated as the background class, otherwise the object belongs to class i.

4.1.2. Private Objects Extraction

After obtaining the objects' information and position, we set a classifier to classify the objects as either private or non-private. Figure 6 is the diagram of private objects extraction part in our framework. As shown in Figure 6, we extract the private objects from all detected objects. In the street view experiment, the private objects can be human faces, car license plates, etc. The non-private objects can be the background, trees, traffic lights.

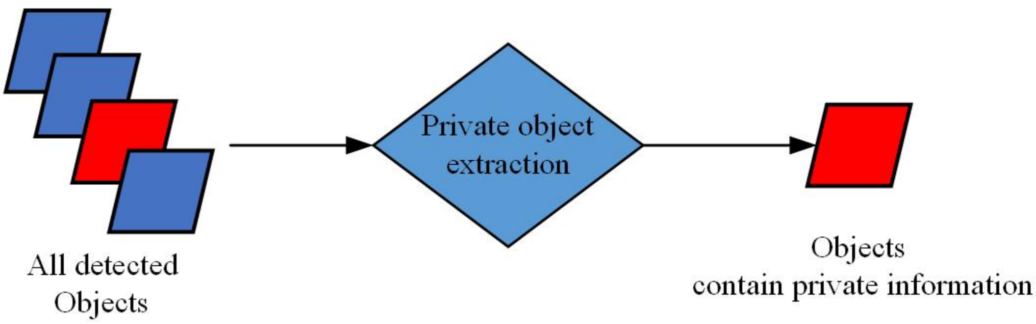

Figure 6. The diagram of private objects extraction part in our image de-identification (DE-ID) framework.

The extraction process is finished by $T(\cdot)$ accordingly as shown in Equation (7).

$$T(\mathbf{O}) = T\left(\begin{array}{cccc|c} x_{p1} & y_{p1} & w_{p1} & h_{p1} & c_{p1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{p\alpha} & y_{p\alpha} & w_{p\alpha} & h_{p\alpha} & c_{p\alpha} \\ \hline x_{np1} & y_{np1} & w_{np1} & h_{np1} & c_{np1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{np\beta} & y_{np\beta} & w_{np\beta} & h_{np\beta} & c_{np\beta} \end{array}\right)$$

$$= \left(\begin{array}{cccc|c} x_{p1} & y_{p1} & w_{p1} & h_{p1} & c_{p1} \\ x_{p2} & y_{p2} & w_{p2} & h_{p2} & c_{p2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{p\alpha} & y_{p\alpha} & w_{p\alpha} & h_{p\alpha} & c_{p\alpha} \end{array}\right) \tag{7}$$

where $(x_{pi}, y_{pi}, w_{pi}, h_{pi})$ and $c_{pi}, i = 1, ..., \alpha$ are the position and class of private objects, and $(x_{npj}, y_{npj}, w_{npj}, h_{npj})$ and $c_{npj}, j = 1, ..., \beta$ are the position and class of non-private objects. So we obtained the private objects' position, class, and pixel information.

The private objects' information is represented as follows:

$$\boldsymbol{P} = T(\boldsymbol{S}|\boldsymbol{C}_p) = \left(\begin{array}{cccc|c} x_{p1} & y_{p1} & w_{p1} & h_{p1} & c_{p1} \\ x_{p2} & y_{p2} & w_{p2} & h_{p2} & c_{p2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{p\alpha} & y_{p\alpha} & w_{p\alpha} & h_{p\alpha} & c_{p\alpha} \end{array}\right) \tag{8}$$

4.2. STEP-II: De-Identification Content Generation

In the second step, we use a content generator $G(\cdot)$ and the differential privacy method to generate the de-identified content. The Algorithm 1 is shown as follows:

Algorithm 1: Image de-identification content generation.

Input: The private image $I_p \in \Re^{n\times m\times 3}$ to de-identify; A pretrained generator $G(\cdot)$.
Output: The de-identified image I_d optimized via $G(\cdot)$
Initialize latent vector ω, differential privacy Laplace noise with Δf and ϵ;
while *not converged* **do**
  $I_p \simeq I'_p = G(\omega^*)$;
end
$I_{pd} = G(\omega^* + Lap(\frac{\Delta f}{\epsilon}))$;

Figure 7 is the diagram of de-identification content generation part in our framework. Firstly, we find the latent vector ω^* of each input image I_p which contains private information. We initialize a latent vector ω and search for an optimized vector ω^* which minimizes the loss function (9). This loss function measures the similarity between the private object image and an image generated by latent vector ω^* [46].

$$\omega^* = \min_{\omega} \mathcal{L}_{percept}(G(\omega), I_p) + \frac{\lambda_{mse}}{N}||G(\omega) - I_p||_2^2 \tag{9}$$

where image $I_p \in \Re^{n\times m\times 3}$ is the input private image. $G(\cdot)$ is the pretrained generator, N is the number of scalars in the image, ω is the latent code to optimize, $\lambda_{mse} = 1$.

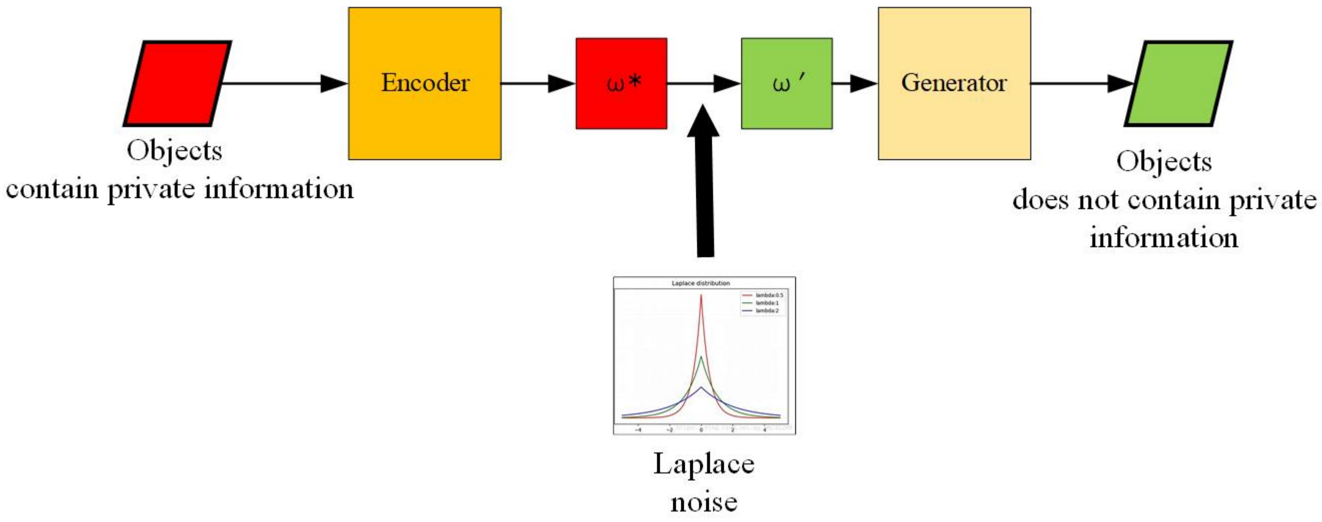

Figure 7. The diagram of de-identification content generation part in our image de-identification (DE-ID) framework.

The loss term $\mathcal{L}_{percept}$ is calculated as follows:

$$\mathcal{L}_{percept}(I_{p1}, I_{p2}) = \sum_{j=1}^{4} \frac{\lambda_j}{N_j} ||F_j(I_{p1}) - F_j(I_{p2})||_2^2, \tag{10}$$

where $I_{p1}, I_{p2} \in \Re^{n\times m\times 3}$ are the input private images, F_j is the feature output of VGG-16 layers conv1_1, conv1_2, conv3_2, conv4_2. N_j is the number of scalars in the jth layer output, $\lambda_j = 1$ for all js are empirically obtained for good performance.

Above step enables the image to be editable. Secondly, after we obtain latent vector ω^* of each private object, we put the Laplace noise on latent vector ω^* and get the new latent vector ω'.

$$\omega' = \omega^* + Lap(\frac{\Delta f}{\epsilon}) \tag{11}$$

Then, we put the new latent vector ω' into the generator $G(\cdot)$ and obtain the de-identified content I_{pd}.

$$I_{pd} = G(\omega^* + Lap(\frac{\Delta f}{\epsilon})) \tag{12}$$

In Equation (12), we used the DP criterion to protect the sensitive information in an image using the Laplace mechanism. Generally speaking, the Laplace mechanism adds a controlled Laplace noise to a query result before returning it to the user. Here, the Laplace noise is sampled from a Laplace distribution, which is shown in Equation (13).

$$Lap(x) = \frac{1}{2b} exp(-\frac{|x|}{b}) \tag{13}$$

The Laplace mechanism can be summarized as

$$M(D) = f(D) + Lap(\frac{\triangle f}{\epsilon}) \tag{14}$$

The Laplace mechanism in Equation (14) indicates that the size of the Laplace noise is related to the sensitivity of query f and the privacy budget ϵ. A larger sensitivity leads to a higher noise. In our method, we use privacy budget ϵ to control our GAN generator to generate the synthetic de-identified content.

4.3. STEP-III: De-Identified Content Replacement

After de-identified content has been generated, we use the generated content to replace the original private object images. The Algorithm 2 is shown as follows:

Algorithm 2: Image protected by de-identification content swapping.

Input: The original image $I \in \Re^{n \times m \times 3}$ contains private objects $P^i, i = 1, 2, ..., N$; de-identified objects in the image: $P_d^i, i = 1, 2, ..., N$
Output: The protected image $I_d \in \Re^{n \times m \times 3}$
for *each* P_d^i ***in*** $\mathbf{P_d}$ **do**
  $P^i \xleftarrow{swapping} P_d^i$
end
$I_d = g(I)$

The original image I contain private objects and we use the de-identified objects generated by our method to replace the private objects. Figure 8 is the diagram of de-identified content replacement part in our framework. We use the images of the de-identify objects (not contain private information) to replace the original private object images (contain private information). After the private image I processed by the above three steps, we finally obtain the de-identified image I_d.

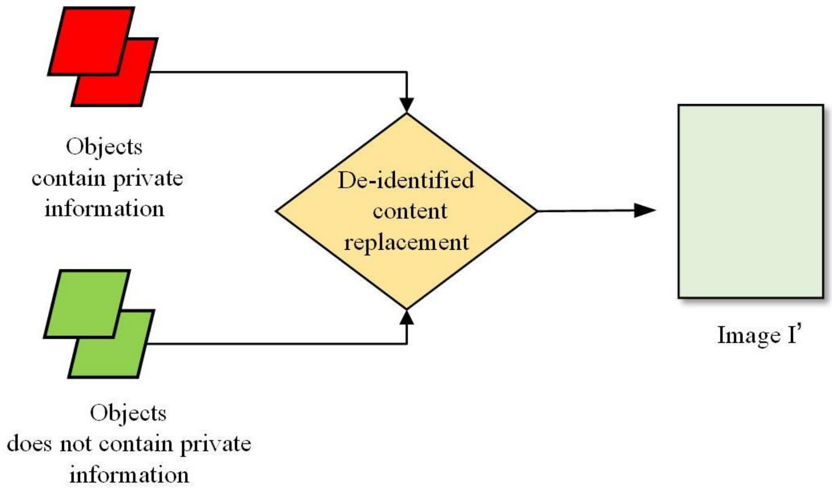

Figure 8. The diagram of de-identified content replacement part in our image de-identification (DE-ID) framework.

5. Experiments and Discussions

In this section, we provide the experiment setup, performance evaluation metrics, street view image protection (include human face privacy protection and car license plate privacy protection) and performance evaluation.

5.1. Experiment Setup

First of all, we set up an experiment database containing the street view images collected by IoMT technology. The street view images contain human faces, car license plates, road signs, traffic lights and more. In these images, the sensitive private information are human faces and car license plates. In our test database, the human faces and car plates are the private objects, and the road sign, the traffic light and background are the non-private objects. We use the camera to collect about 4000 typical street view images as the test database.

5.2. Performance Evaluation Metrics

5.2.1. Privacy Metrics

Confidence Score. In the privacy protection metric for a human face, we use the open-source "face recognition" platform to evaluate the confidence in face privacy. This platform was built using dlib's state-of-the-art face recognition which was built with deep learning. The model has an accuracy of 99.38% on the Labeled Faces in the Wild benchmark. The output of the platform is the facial distance between each unrecognized face and the recognized face. By setting the corresponding threshold, the distance metric can judge whether the face is protected. This means after the face photo is processed by our method, we can determine whether the general third-party platform still considers it be the same person. The default threshold is 0.3.

Distance. In the privacy protection metric for the car license plate, as the license plate is a set of characters, we believe that the distance between the original license plate and the processed license plate is the privacy metric. In the experiment, we set the threshold of the car license plate as 3. This means that the sensitive information on the license plates is protected when the distance is greater than 3.

5.2.2. Image Utility Metrics

Quantitative judgment is necessary for the degree of modification between the original image and the protected image. So, we use several metrics to calculate the degree of modification, including L_0, L_2, ALD_p, the structural similarity index (SSIM), and the difference value hash(Dhash). Deciding when there are two images: processed image I^a and original image I, the utility image metrics are:

L_0 calculates the number of changed pixels.

$$L_0 = num(I^a, I) \tag{15}$$

where num is calculated as the number of pixels changed between I^a and I.

L_2 calculates the Euclidean distance between the original image and the protected image.

$$L_2 = ||I^a - I||_2 = \sqrt{\sum_{i=1}^{N}(I_i^a - I_i)^2} \tag{16}$$

ALD calculates the average L distance between the images.

$$ALD_p = \frac{1}{n}\sum_{i=1}^{n}\frac{\|I_i^a - I_i\|_p}{\|I_i\|_p} \tag{17}$$

SSIM is the common method to evaluate the similarity between the original image and the protected image.

$$SSIM(I^a, I) = \frac{1}{n}\sum_{i=1}^{n} SSIM(I_i^a, I_i) \tag{18}$$

Dhash uses the difference hash to evaluate the degree of modification where the smaller the value, the better.

$$Dhash(I^a, I) = hash(I^a) - hash(I) \tag{19}$$

5.3. Street View Image Protection

5.3.1. Human Face Privacy Protection

The human face is the most sensitive information in IoMT images, which can reveal a person's identity. Therefore, we use our method to protect facial privacy in the street view experiment.

The algorithm to de-identify the facial image is shown in Algorithm 3.

Algorithm 3: Facial image de-identification.

Input: A human face image $I \in \Re^{n \times m \times 3}$ to protect; a pretrained generator $G(\cdot)$
Output: The de-identify facial image I_d.
Initialize latent code $\omega^* = \omega$;
while *not converged* **do**
  $I \simeq I' = G(\omega^*)$;
end
$I_d = G(\omega^* + Lap(\frac{\Delta f}{\epsilon}))$

Firstly, we use Mask-RCNN to extract facial images I from the experiment street view images.

Secondly, we initialize latent vector ω and use the loss function to find latent vector ω^* of human face I.

Thirdly, we put the Laplace noise on latent vector ω^* and use generator $G(\cdot)$ to generate the de-identified facial image.

$$I_d = G(\omega^* + Lap(\frac{\Delta f}{\epsilon})) \quad (20)$$

Finally, we swap the de-identified facial image for the original facial image. In this step, we use Dlib, which is a toolbox in OpenCV based on key-point face detection, to obtain the 68 key points of the faces and use seamless cloning to swap the faces. The face swapping algorithm can transfer the input facial features to the target face without being obtrusive.

Figure 9 is an example of the original human face image and the human face generated by GAN with no modification.

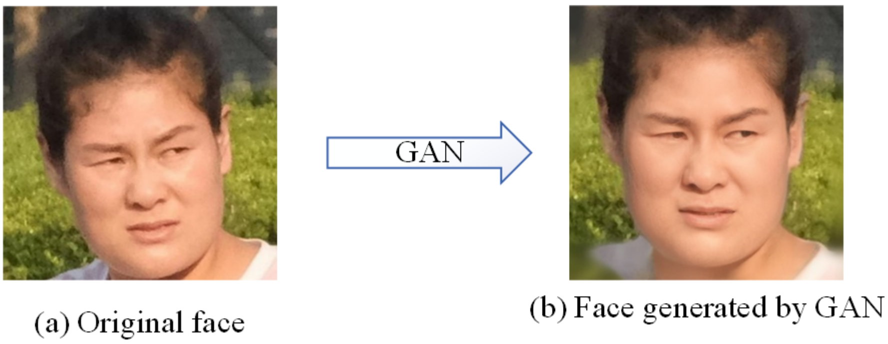

Figure 9. The original face image projected into StyleGAN: (**a**) Original face. (**b**) Face generated by GAN.

The de-identified example result is shown in Figure 10. Intuitively speaking, a larger Laplace noise results in the generation of a very different face compared with the original photo of the face. In our experiments, we use Laplace noise parameter ϵ to control the distance between the de-identified facial image and original facial image. In addition, we use the open-source "face recognition" platform to determine if the synthetic facial image and the original facial image represent the same person.

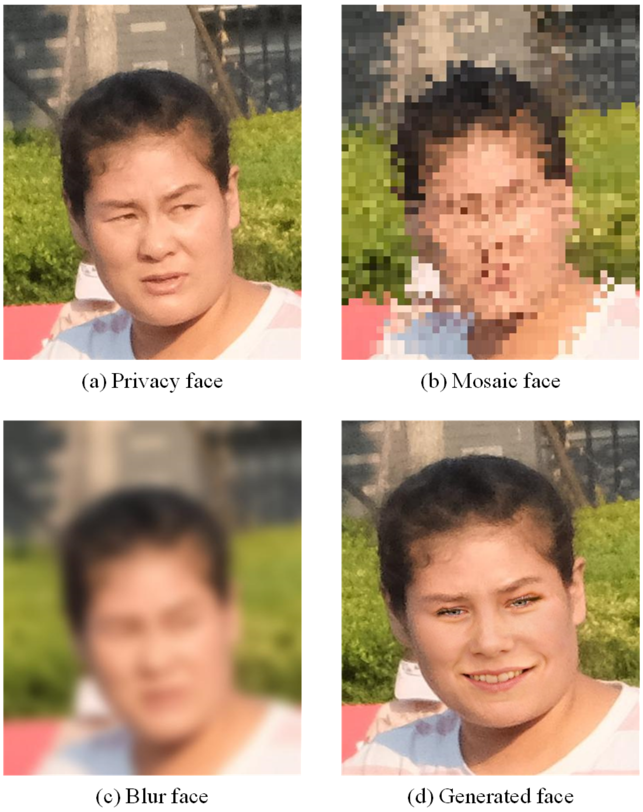

Figure 10. Face images comparison: (**a**) face in street view, (**b**) mosaic methods, (**c**) blur method, and (**d**) our method.

5.3.2. Facial Privacy Protection Discussion

There are currently many methods for face swap and generation, the main methods include DeepFakes [47], Face2Face [48], FaceSwap [49], and NeuralTextures [50], etc. These methods can well swap the source face to achieve the purpose of changing the source face. However, in 2019, Andreas et al. propose an automated benchmark for facial manipulation detection called FaceForensics++ [51] which can detect manipulated facial images. This benchmark can easily detect the manipulated facial images with high efficiency. However, the previous methods discussed the application and effects of face swapping and how to detect manipulated facial images. We are more concerned about the privacy protection of IoMT images. In our method, we propose a framework that uses GAN and DP to protect the multiobject privacy of IoMT images. Different from other face swap methods, our method can not only protect facial information but also protect other private information, such as license plates. Our replacement content is generated by GAN, and DP technology is applied to control the generated content.

Figure 10 shows the original photo of the face, the mosaic face, the blurred face, and the new facial image generated by our method. It can be seen that it is not easy for either a human or a machine to recognize the de-identified generated facial image in Figure 10d compared to Figure 10a. It is worth mentioning that if the protected person sees that his\her facial information is replaced by computer-generated content, he should feel at ease because his\her personal facial information has been de-identified.

To protect the privacy of IoMT images, just replacing faces is not enough to protect privacy, multiobject privacy needs to be protected. So we set a framework that can protect multiobject privacy. There are many private contents in IoMT images that need to be protected. In our method, we added car license plates as another type of privacy that needs to be protected.

5.3.3. Car License Plate Privacy Protection

Car license plates are another type of sensitive objects in IoMT images. We use Chinese car license plates as our experimental objects. The car license plates should be generated according to the rules enforced by the vehicle management authority. The rules of a valid Chinese car license plate are: (1) the first character is a Chinese character, representing a province; (2) the second symbol is an English letter; (3) the last five symbols form a random string of letters and numbers, and (4) the background of a license plate is dark blue.

After obtaining the car license plate images from the street images, we use OCR to recognize the characters and symbols on the car license plates, and then map the car license plate into a sequence of numbers. As previously mentioned, the first character will be one of 31 Chinese province abbreviation characters (with the exception of special districts). Because the first Chinese character represents location information, we map this into two-digit numbers 00–30 based on the sorted distances from each province to the capital city Beijing. The mapping table for the first character is shown in Table A1.

Next, the numerical values 0–9 are translated into two-digit codes 00–09, and the English symbols are translated into two-digit codes 10–33. For example, a car plate "Beijing A132B3" will be mapped to a sequence of numbers "00 100103021103". After we translate each car plate into a sequence of numbers, we add Laplace noise onto the number sequence and obtain a synthetic number sequence satisfying DP. In Laplace noise generation, we let $\Delta f = 1$ and control ϵ to generate the Laplace noise. For example, if we add a random Laplace noise on the above car plate "00 100103021103", we obtain a perturbed sequence as "03 130214231502", which can be translated to a synthetic car plate "Hebei D2ENF2". The above example is illustrated in Figure 11. There is a cyclic shift if the Laplace noise results in the value being out of the bounds, e.g., the province code > 33.

Then, we use the generator to generate a synthetic car plate image according to the car plate code. Finally, we swap the car plate with the synthetic car plate image. The synthetic car plate is protected by the DP criterion.

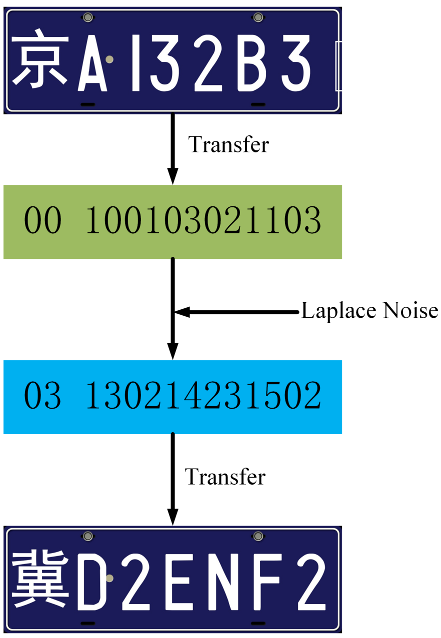

Figure 11. A new car plate content created by DP.

In the car license plate number transfer, the larger the noise, the longer distance car number is generated. For example, if a province name is Jilin on a car plate, the province codes should be generated for Jilin based on the distance from the other provinces to Jilin.

5.3.4. Car License Plate Privacy Protection Discussion

Car license plates as another type of sensitive objects in IoMT images. We choose car license plate as another type of multiobject privacy protection representative. The framework we proposed can add more types of private objects, which is highly scalable. In future work, We will add more types of private objects.

It is important to note that the replacement of the private content in an image is not simply a copy-and-paste job. Instead, it needs to transform the synthetic content by generating an image that fits into the original image area with the correct orientation. Therefore, the synthetic image is generally not perceptible to human eyes.

Our method uses the synthetic DP car plate to protect the private car plate information. As shown in Figure 12, we can see that the car plate is smoothly replaced by our synthetic car plate.

(a) Original street view image (b) Our method

Figure 12. A typical Chinese car plate swap to protect the private information in street view image: (**a**) Original street view image. (**b**) Our method.

5.4. Performance Evaluation

5.4.1. Privacy Protection Metrics

In this part, we calculate the distance between the original private image and the protected image to measure the degree of privacy protection.

For a human face, the average distance between the same person is 0.12, which has a confidence score of 88. After using our processing method, the average facial distance is 0.45 with a confidence score of 55, which is over the threshold of the confidence score of 70. This experiment result means our method can remove the identity of the human face, which means our method can protect the privacy of facial image.

For car license plates, because the license plates are strings, their distances are integers. In the experiment, the distance between the same license plate is 0. After using our processing method, the distance is 3, so the sensitive information in the image of the car license plate is protected.

5.4.2. Image Utility Metrics

In this part, we set an automatic evaluation module to calculate the degree of image modification using different metrics through L_0, L_2, ALD_p, $SSIM$, and $Dhash$. We compare our method with the blur and mosaic methods. As shown in Figure 13, the blur and mosaic methods remove the sensitive private area. However, a human can easily notice the blur and mosaic in the image. Hence, a computer can recover the information from the processed image [52–54].

In our method, we control the generator to generate the de-identified content image with DP Laplace noise. The de-identified images make it very difficult for human eyes and computer vision detection methods to detect the differences and obtain the private information for sensitive private objects. The results of the street view image are shown in Figure 13. It can be seen that a human and a computer can easily detect the sensitive information in the unprotected street view image in Figure 13a. In Figure 13b,c, the algorithm cannot detect the face or the car license plate number after being blurred, but a human can easily see there is a blur or mosaic in the image. In relation to Figure 13d, both a computer algorithm and a human can detect the changed sensitive information, so neither a human nor a computer can not see the real sensitive information of the face and the car plate. Hence, our method protects the private information in the image.

(a) Original street view (b) Mosaic method

(c) Blur method (d) Our method

Figure 13. The result of four street view images: (**a**) unprotected image. (**b**) image processed with blur, (**c**) image processed with mosaic, (**d**) image processed with our method.

Next, we use metrics to evaluate the efforts of our method. Table 1 shows the performance of our method, blur, and mosaic. The metrics are DHash, SSIM, L_0, L_2 and ALD_p. The blur and mosaic are modified to change the sensitive area in our experiment images.

First, compared with the other methods, our method changes the minimum number of pixels to protect the private part of the image. For Dhash, our method is better than the others. Compared with blur and mosaic, our method decreases the Dhash metric by

95.02% and 95.2%. Our method is better than the others in SSIM metric by 1.17% and 1.67%. Our method decreases the L_0 metric by 73.6% and 72.97% compare with blur and mosaic. In L_2 metric, our method decrease the L_2 metric by 86.25% and 25.99%. In ALD_p metric, our method's result is higher than blur and mosaic, which is 160.65% and 98.85%.

This shows that our method is better than the other two methods in metrics: SSIM, Dhash and L_0. However, the results show that in metrics: L_2 and ALD_p, our method is not the best. After analysis, we found that L_2 and ALD_p are more suitable in big area modification. These metrics are not sensitive to minor modifications.

We use the facial image as an example to show the metrics' results in the minor modification in a small area. We choose 4000 face swap images to analyze and the results are shown in Table 2. In Dhash metric results, compared with blur and mosaic, our method's result decreases by 96.68% and 96.97%. In SSIM metric results, our method increases by 50.67% and 102.24% compared with blur and mosaic. In L_0 metric result, our method decreases by 76.55% and 76.84% than blur and mosaic. In L_2 metric result, our method decreases by 64.93% and 81.08% than blur and mosaic. In ALD_p metric result, compared to blur and mosaic, our method decreases by 65.11% and 79.68%. As we can see, our method is the best for all metrics in the evaluation of minor modification in a small area.

Table 1. Average result of 4000 street view images with the metrics: Dhash, SSIM, L_0, L_2, ALD_p.

Methods	Original	Blur	Mosaic	Our Methods
Dhash	0	12,873.65	13,370.19	641.71
SSIM(10^{-2})	100	98.18	97.70	99.33
$L_0(10^2)$	0	1692.25	1652.57	446.74
L_2	0	9983.06	14,757.19	18,593.41
$ALD_p(10^{-2})$	0	3.99	5.23	10.4

Table 2. Average result of 4000 facial images with the metrics: Dhash, SSIM, L_0, L_2, ALD_p.

Methods	Original	Blur	Mosaic	Our Methods
Dhash	0	4047.80	4427.79	134.25
SSIM(10^{-2})	100	64.63	48.15	97.38
$L_0(10^2)$	0	1009.4	1022.25	236.72
L_2	0	5832.96	10,812.2	2045.48
$ALD_p(10^{-2})$	0	16.68	28.64	5.82

6. Conclusions

In this paper, we propose a new image privacy protection method based on GAN and DP. Our method can protect the sensitive private information contained in IoMT images. We use the deep neural network to identify the private data in images and de-identify this with the GAN-based content. Compared with traditional blur and mosaic methods, the proposed method can protect the sensitive information in image data and avoid privacy leakage. The experiment results of the collected IoMT image data show that our privacy protection method can protect privacy with high efficiency and controllability. In future work, we will study the privacy protection of IoMT videos and improve the real-time nature of our method. We will also add more types of private objects into our framework and propose a higher effectively privacy protection method for the privacy of IoMT images.

Author Contributions: Conceptualization, J.Y. and B.L.; methodology, J.Y.; software, J.Y. and H.X.; validation, J.Y. and H.X.; formal analysis, J.Y.; investigation, J.Y.; resources, J.Y.; data curation, J.Y.; writing—original draft preparation, J.Y.; writing—review and editing, B.L., Y.W., M.D., and S.Z.; visualization, J.Y.; supervision, B.L. and S.Z.; project administration, Y.W. and J.Y.; funding acquisition, S.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Science and Technology on Complex Electronic Simulation Laboratory Foundation under grant Number DXZT-JC-ZZ2017-005, the National Natural Science Foundation of China under grant Number 61802080, the Education Bureau of Guangzhou Municipality Higher Education Research Project under grant Number 201831827 and the Guangzhou University Research Project under grant Number RQ2020085.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki, and the protocol was approved by the Ethics Committee of Space Engineering University.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy protection.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. The transfer two-digit code based on distance between Beijing and each province of China.

Province Name	Distance to Beijing (km)	2-Digit Code
Beijing	0	00
Tianjin	96.07188	01
Hebei	239.4603	02
Shandong	356.9375	03
Shanxi	407.3106	04
Neimengu	424.5428	05
Henan	620.2232	06
Liaoning	630.724	07
Jiangsu	860.7032	08
Jilin	867.213	09
Ningxia	884.2019	10
Anhui	897.8403	11
Shanxi	907.8513	12
Hubei	1041.318	13
Shanghai	1041.987	14
Heilongjiang	1056.846	15
Zhejiang	1102.843	16
Gansu	1184.73	17
Jiangxi	1242.833	18
Hunan	1316.041	19
Qinghai	1340.82	20
Chongqing	1419.309	21
Sichuan	1505.931	22
Fujian	1527.525	23
Guizhou	1729.627	24
Guangdong	1856.641	25
Guangxi	2047.263	26
Yunnan	2068.306	27
Hainan	2249.545	28
Xinjiang	2433.955	29
Xizang	2559.149	30

References

1. EU. The EU General Data Protection Regulation. 2019. Available online: https://eugdpr.org/ (accessed on 19 July 2019).
2. EU. What is Considered Personal Data? 2019. Available online: https://gdpr.eu/eu-gdpr-personal-data/ (accessed on 19 July 2019).
3. Mannan, M.; van Oorschot, P.C. Privacy-enhanced sharing of personal content on the web. In Proceedings of the 17th International Conference on World Wide Web—WWW '08, Beijing, China, 21–25 April 2008; ACM Press: Beijing, China, 2008; p. 487. [CrossRef]

4. Vyas, N.; Squicciarini, A.C.; Chang, C.C.; Yao, D. Towards automatic privacy management in Web 2.0 with semantic analysis on annotations. In Proceedings of the 5th International ICST Conference on Collaborative Computing: Networking, Applications, Worksharing, Washington, DC, USA, 11–14 November 2009; doi:10.4108/ICST.COLLABORATECOM2009.8340. [CrossRef]
5. Wang, N.; Xu, H.; Grossklags, J. Third-party apps on Facebook: Privacy and the illusion of control. In Proceedings of the 5th ACM Symposium on Computer Human Interaction for Management of Information Technology—CHIMIT '11, Cambridge, MA, USA, 4–5 December 2011; pp. 1–10. [CrossRef]
6. Squicciarini, A.C.; Xu, H.; Zhang, X.L. CoPE: Enabling collaborative privacy management in online social networks. *J. Am. Soc. Inf. Sci. Technol.* **2011**, *62*, 521–534. [CrossRef]
7. Liu, B.; Ding, M.; Shaham, S.; Rahayu, W.; Farokhi, F.; Lin, Z. When Machine Learning Meets Privacy: A Survey and Outlook. *arXiv* **2020**, arXiv:2011.11819.
8. Viola, P.; Jones, M.J. Robust real-time face detection. *Int. J. Comput. Vis.* **2004**, *57*, 137–154. [CrossRef]
9. Wen, Y.; Liu, B.; Xie, R.; Zhu, Y.; Cao, j.; Song, L. A Hybrid Model for Natural Face De-Identification with Adjustable Privacy. In Proceedings of the 2020 IEEE Visual Communications and Image Processing (VCIP), Virtual Conference, 1–4 December 2020.
10. Tonge, A.; Caragea, C. Image privacy prediction using deep neural networks. *ACM Trans. Web (TWEB)* **2020**, *14*, 1–32. [CrossRef]
11. Yu, J.; Zhang, B.; Kuang, Z.; Lin, D.; Fan, J. iPrivacy: Image Privacy Protection by Identifying Sensitive Objects via Deep Multi-Task Learning. *IEEE Trans. Inf. Forensics Secur.* **2017**, *12*, 1005–1016. [CrossRef]
12. Uittenbogaard, R.; Sebastian, C.; Vijverberg, J.; Boom, B.; Gavrila, D.; de With, P.H.N. Privacy Protection in Street-View Panoramas Using Depth and Multi-View Imagery. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 10573–10582.
13. Liu, Y.; Zhang, W.; Yu, N. Protecting Privacy in Shared Photos via Adversarial Examples Based Stealth. *Secur. Commun. Netw.* **2017**, *2017*, 1897438. [CrossRef]
14. Liu, B.; Xiong, J.; Wu, Y.; Ding, M.; Wu, C.M. Protecting Multimedia Privacy from Both Humans and AI. In Proceedings of the 2019 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Jeju, Korea, 5–7 June 2019; pp. 1–6.
15. Liu, B.; Ding, M.; Zhu, T.; Xiang, Y.; Zhou, W. Adversaries or allies? Privacy and deep learning in big data era. *Concurr. Comput. Pract. Exp.* **2019**, *31*, e5102. [CrossRef]
16. Xue, H.; Liu, B.; Ding, M.; Song, L.; Zhu, T. Hiding Private Information in Images From AI. In Proceedings of the 2020 IEEE International Conference on Communications (ICC), Dublin, Ireland, 7–11 June 2020.
17. McPherson, R.; Shokri, R.; Shmatikov, V. Defeating image obfuscation with deep learning. *arXiv* **2016**, arXiv:1609.00408.
18. Tang, L.; Ma, W.; Grobler, M.; Meng, W.; Wang, Y.; Wen, S. Faces are Protected as Privacy: An Automatic Tagging Framework Against Unpermitted Photo Sharing in Social Media. *IEEE Access* **2019**, *7*, 75556–75567. [CrossRef]
19. Pesce, J.P.; Casas, D.L. Privacy Attacks in Social Media Using Photo Tagging Networks: A Case Study with Facebook. In Proceedings of the 1st Workshop on Privacy and Security in Online Social Media, Lyon, France, 17 April 2012; pp. 1–8.
20. The New York Times. San Francisco Bans Facial Recognition Technology. 2019. Available online: https://www.nytimes.com/2019/05/14/us/facial-recognition-ban-san-francisco.html (accessed on 19 July 2019).
21. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 24–27 June 2014; pp. 580–587. [CrossRef]
22. Hariharan, B.; Arbelaez, P.; Girshick, R.; Malik, J. Hypercolumns for object segmentation and fine-grained localization. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 447–456. [CrossRef]
23. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Las Condes, Chile, 11–18 December 2015; pp. 1440–1448.
24. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–10 December 2015; pp. 91–99.
25. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
26. He, K.; Gkioxari, G.; Dollar, P.; Girshick, R. Mask R-CNN. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2980–2988.
27. Efros, A.; Leung, T. Texture synthesis by non-parametric sampling. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, 20–25 September 1999; Volume 2, pp. 1033–1038. [CrossRef]
28. Barnes, C.; Shechtman, E.; Finkelstein, A.; Goldman, D.B. PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing. *ACM Trans. Graph.* **2009**, *28*, 24. [CrossRef]
29. Pnevmatikakis, E.A.; Maragos, P. An inpainting system for automatic image structure—Texture restoration with text removal. In Proceedings of the 2008 15th IEEE International Conference on Image Processing, San Diego, CA, USA, 12–15 October 2008; pp. 2616–2619. [CrossRef]
30. Bertalmio, M.; Vese, L.; Sapiro, G.; Osher, S. Simultaneous Structure and Texture Image Inpainting. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Madison, WI, USA, 18–20 June 2003.

31. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Networks. *arXiv* **2014**, arXiv: 1406.2661.
32. Klambauer, G.; Unterthiner, T.; Mayr, A.; Hochreiter, S. Self-normalizing neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 971–980.
33. Mao, X.; Li, Q.; Xie, H.; Lau, R.; Wang, Z.; Smolley, S. Least Squares Generative Adversarial Networks. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2813–2821.
34. Xiong, W.; Yu, J.; Lin, Z.; Yang, J.; Lu, X.; Barnes, C.; Luo, J. Foreground-Aware Image Inpainting. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 5833–5841.
35. Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; Courville, A.C. Improved training of wasserstein gans. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5767–5777.
36. Karras, T.; Laine, S.; Aila, T. A Style-Based Generator Architecture for Generative Adversarial Networks. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 4396–4405.
37. Meng, W.; Li, W.; Wang, Y.; Au, M.H. Detecting insider attacks in medical cyber–physical networks based on behavioral profiling. *Future Gener. Comput. Syst.* **2020**, *108*, 1258–1266. [CrossRef]
38. Wang, Y.; Meng, W.; Li, W.; Li, J.; Liu, W.X.; Xiang, Y. A fog-based privacy-preserving approach for distributed signature-based intrusion detection. *J. Parallel Distrib. Comput.* **2018**, *122*, 26–35. [CrossRef]
39. Ma, L.; Liu, X.; Pei, Q.; Xiang, Y. Privacy-Preserving Reputation Management for Edge Computing Enhanced Mobile Crowdsensing. *IEEE Trans. Serv. Comput.* **2019**, *12*, 786–799. [CrossRef]
40. Li, L.; Liu, J.; Cheng, L.; Qiu, S.; Wang, W.; Zhang, X.; Zhang, Z. CreditCoin: A Privacy-Preserving Blockchain-Based Incentive Announcement Network for Communications of Smart Vehicles. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 2204–2220. [CrossRef]
41. Wang, M.; Zhu, T.; Zhang, T.; Zhang, J.; Yu, S.; Zhou, W. Security and privacy in 6G networks: New areas and new challenges. *Digit. Commun. Netw.* **2020**, *6*, 281–291. [CrossRef]
42. Sweeney, L. k-anonymity: A model for protecting privacy. *Int. J. Uncertain. Fuzziness-Knowl. Syst.* **2002**, *10*, 557–570. [CrossRef]
43. Machanavajjhala, A.; Kifer, D.; Gehrke, J.; Venkitasubramaniam, M. L-diversity: Privacy beyond k-anonymity. *ACM Trans. Knowl. Discov. Data* **2007**, *1*, 3-es. [CrossRef]
44. Li, N.; Li, T.; Venkatasubramanian, S. t-closeness: Privacy beyond k-anonymity and l-diversity. In Proceedings of the 2007 IEEE 23rd International Conference on Data Engineering, Istanbul, Turkey, 15–20 April 2017; pp. 106–115.
45. Dwork, C. Differential privacy. In Proceedings of the 33rd International Conference on Automata, Languages and Programming-Volume Part II, Beijing, China, 15–20 May 2006; Springer: Berlin/Heidelberg, Germany, 2006; pp. 1–12.
46. Abdal, R.; Qin, Y.; Wonka, P. Image2stylegan: How to embed images into the stylegan latent space? In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 4432–4441.
47. Nguyen, T.; Nguyen, C.; Nguyen, D.; Nahavandi, S. Deep learning for deepfakes creation and detection. *arXiv* **2019**, arXiv:1909.11573.
48. Thies, J.; Zollhofer, M.; Stamminger, M.; Theobalt, C.; Nießner, M. Face2face: Real-time face capture and reenactment of rgb videos. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 2387–2395.
49. Korshunova, I.; Shi, W.; Dambre, J.; Theis, L. Fast face-swap using convolutional neural networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 3677–3685.
50. Thies, J.; Zollhöfer, M.; Nießner, M. Deferred neural rendering: Image synthesis using neural textures. In *ACM Transactions on Graphics*; ACM: New York, NY, USA, 2019; pp. 1–12.
51. Rossler, A.; Cozzolino, D.; Verdoliva, L.; Riess, C.; Thies, J.; Nießner, M. Faceforensics++: Learning to detect manipulated facial images. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 1–11.
52. Kupyn, O.; Budzan, V.; Mykhailych, M.; Mishkin, D.; Matas, J. DeblurGAN: Blind Motion Deblurring Using Conditional Adversarial Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 8183–8192.
53. Nah, S.; Kim, T.H.; Lee, K.M. Deep Multi-scale Convolutional Neural Network for Dynamic Scene Deblurring. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 257–265.
54. Menon, S.; Damian, A.; Hu, S.; Ravi, N.; Rudin, C. PULSE: Self-Supervised Photo Upsampling via Latent Space Exploration of Generative Models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–18 June 2020; pp. 2437–2445.