Hindawi
Security and Communication Networks
Volume 2022, Article ID 4466426, 11 pages
https://doi.org/10.1155/2022/4466426

Research Article

Domain Transferred Image Recognition via Generative Adversarial Network

Haoqi Hu, Sheng Li, Zhenxing Qian, and Xinpeng Zhang

Fudan University, Shanghai, China

Correspondence should be addressed to Zhenxing Qian; zxqian@fudan.edu.cn

Received 23 February 2022; Accepted 5 April 2022; Published 26 April 2022

Academic Editor: Beijing Chen

Recent studies have demonstrated that neural networks exhibit excellent performance in information hiding and image domain transfer. Considering the tremendous progress that deep learning has made in image recognition, we explore whether neural networks can recognize the imperceptible image in the transferred domain. Our target is to transfer natural images into images that belong to a different domain, while at the same time, the attribute of natural images can be recognized on domain transferred images directly. To address this issue, we proposed domain transferred image recognition to achieve image recognition directly on the transferred images without the original images. In our proposed system, a generator is designed for the domain transfer and a recognizer is responsible for image recognition. To be flexible for the natural image restoration in some cases, we also incorporate an additional generator in our method. In addition, a discriminator will play an indispensable role in the image domain transfer. Finally, we demonstrate that our method can successfully identify the natural images on transferred images without access to original images.

1. Introduction

Recently, there have emerged numerous methods for privacy protection-awareness [1–3]. At the same time, deep learning has made great breakthroughs in speech, image, and text recognition [4–6]. However, training these networks requires a large amount of data, which makes some giant companies such as Google, Microsoft, and Amazon or personalized customization organizations which try every means to collect personal data of their users for training deep models [7]. Despite of great performance of these well-trained deep networks, they bring huge privacy risks [8, 9].

Most of small businesses or individuals use the cloud services provided by giant companies for deep learning tasks since they are limited to the local storage capacity and GPU resources. However, the data collected by these organizations can be reused repeatedly, making users difficult to delete. Besides, these sensitive data may contain unique personal identical information such as faces and voices, which inevitably bring risks when stolen by malicious attackers and used for illegal benefits [10, 11].

To securely perform image recognition, information hiding, which conceals important secret information in the carrier (image, video, audio, etc.), can solve the issue of privacy leakage elegantly and flexibly [12, 13]. However, information hiding mainly focuses on protecting secret information from being leaked during transmission. As shown in Figure 1, when image recognition is required, it has to restore the secret images, which increases the risk of information leakage. Moreover, to hide abundant secret information, it is necessary to select an appropriate carrier with large redundant room, which is time-consuming and will inevitably increase the risk of information leakage once the carrier is intercepted. To this end, we propose a method that can achieve image recognition in the transferred domain directly, which can recognize the attributes of secret images on transferred images.

Specially, our detailed application scenario for real world is shown in Figure 2; a giant company deploys an image recognition service in the cloud for profits. To avoid data leakage of users, we train a series of models ahead to reach this goal. In the real service deployment, the service provider owns/deploys only the well-trained recognizer in its cloud, and the domain transfer generator is deployed in a trustworthy party.

Therefore, a user uploads a secret image to the trustworthy party to transfer his/her image into another image

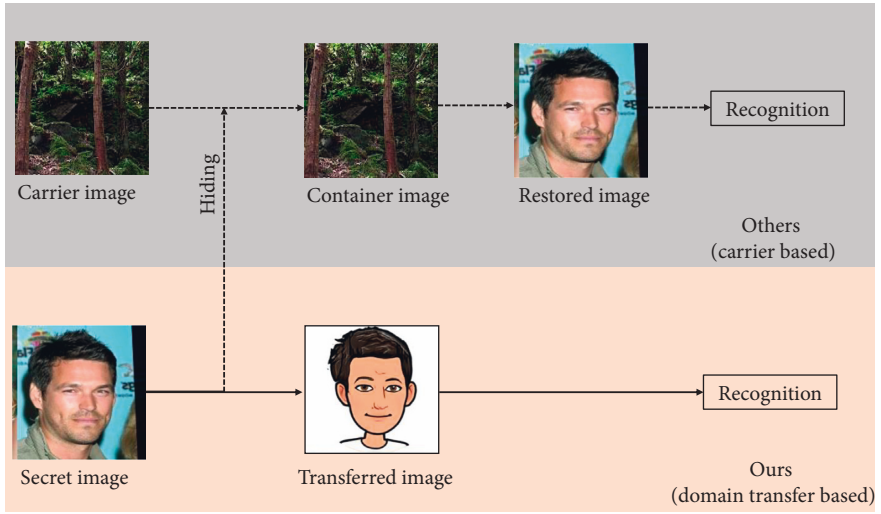

FIGURE 1: The comparison of application processes. For a system that requires image recognition in the cloud server, to protect the image content privacy, other methods based on information hiding will first select a textured carrier/cover image and then apply a special embedding algorithm to hide the secret image in a carrier image and send it to the cloud end. The receiver will apply the corresponding extracting method to obtain the secret image for recognition. Differently, our method is free from carrier image by transferring the secret image to another domain and the recognition process can be performed on the transferred domain directly.

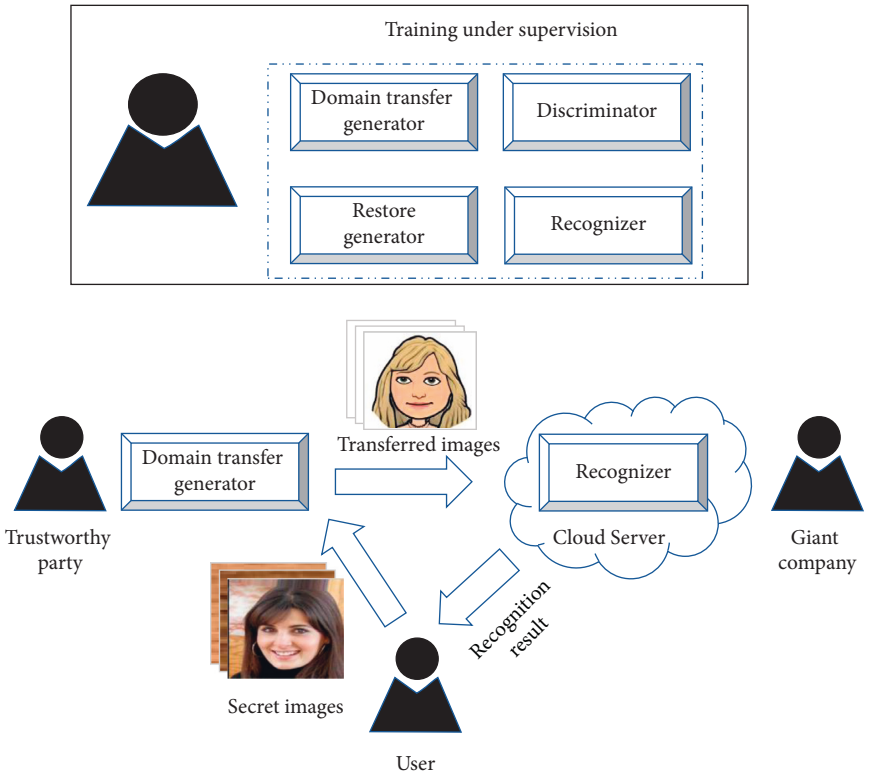

FIGURE 2: The conception of our system. The models are trained under supervision, where the domain transfer generator is used to protect the content privacy of the user's secret images and a recognizer network which is used to classify secret images in the transferred domain. In the cloud recognition application scenario, the domain transfer generator is handed over to a trusted party, and the recognizer is controlled by an organization which is deployed on a cloud service.

which has different content/style and then delivers the transferred image to the cloud for image recognition.

To perform image recognition in the transferred domain, we designed a transfer generator to accomplish the process of domain transfer. Same as the classic generative adversarial networks, an indispensable discriminator aids the generator to generate a high-quality image. Besides, a classifier is responsible for the image recognition task. Although our system is able to recognize the attributes of the original images through the transferred images directly, to facilitate

the original image restoration in some cases, we also incorporate an additional restore generator to recover the secret image in our method. In summary, our contributions are three-fold:

(i) To achieve image recognition as well as avoid private information leakage issue, we proposed a framework to perform image recognition directly on domain transferred images

(ii) Our image recognition method is performed on the transferred domain, which decreases the exposure risk of the source image and omits the process of carrier selection

(iii) Experiments demonstrate the availability of our method in terms of classification accuracy and visual effect of the transferred images

2. Related Work

In this section, we will first review some secure inference and visual information protection strategies.

2.1. Encryption. Encryption technique is used for data privacy protection. Its basic target is to hide the content of the data, which encrypts the raw data into the ciphertext data and decrypts it back to the original version with the corresponding decryption algorithm.

Homomorphic encryption proposed by Craig [14] can support arbitrary computations on the encrypted data and the final calculation result can be obtained after decryption. However, this technique, which is calculated in the encrypted domain, is computationally expensive compared to plaintext calculation. For example, recently, Sanyal [15] proposed a homomorphic encryption-based image classification algorithm to protect image privacy. However, it takes nearly 2 hours to classify encrypted MNIST images on a 16-core workstation, which is impractical in reality. Moreover, the encrypted images can raise the awareness of attackers due to its unreasonable ciphertext.

Secure multiparty computation is an important branch of cryptography, which aims to solve the problem of collaborative computing that protects privacy among a group of untrusted parties. Implementations of predicting encrypted data based on secure multiparty computing have flourished [16, 17]. However, these methods require the data owner to encrypt the inputs and constantly interact and communicate with each other. Besides, as mentioned in [18], most of the existing multiparty computation-based secure inferences rely on customized protocols that are highly optimized for particular activation functions. For example, XONN [19] is currently the most efficient solution for 2-party protocol, but XONN only works with Sign as the activation function. Implementing exponential function (Sigmoid) or max function (ReLU) requires heavy computations and communications in SMPC-based solutions.

2.2. Information Hiding. Information hiding is one of the most important ways to protect secret data, which has been well researched in the past decades [20–22]. This technique can be roughly classified into two categories: digital watermarking and steganography. Recently, many deep learning algorithms related to information hiding have been developed. Usually, digital watermarking technology hides a particular bit string in inconspicuous places to protect the copyright of images, models, etc. Uchida et al. [23] embed a watermark in model parameters using a regularizer to protect the intellectual property of trained models, with the performance of trained model hardly affected. Rouhani et al. [24] embed watermarking in the weight distribution of convolutional layers in trained models to protect deep learning models. Baluja [25] successfully hides a full-size color image into another image of the same size based on a deep image encoder and decoder network to realize image steganography. However, these methods focus on protecting the security of hidden information during transmission. These methods have to go through the process of extracting hidden secret information when using these information, which may lead to hidden secret information leakage after restoration.

2.3. Image Domain Transfer. Image domain transfer refers to the process of generating another image according to one image, that is, transfer one image from one domain to another. Pix2pix proposed by Isola et al. [26] provides a concise and elegant general framework for solving a series of image domain transfer tasks, and the author proves that the method has good performance in tasks such as segmentation map to street view map, grayscale map to color map, and clothing outline sketch to color map. In addition to supervised image domain tasks (with paired training samples), for unsupervised image domain transfer tasks without paired images as training samples, Zhu et al. [27] designed CycleGAN to complete the image domain transfer process from one dataset to another dataset. Moreover, StarGAN [28] proposed by Choi et al. can complete various attribute conversions of face images, such as gender, age, skin color, and emotions, and Tang et al. [29] proposed a method that can transfer an image from a source to a target domain guided by controllable structures. However, these GAN-based image domain transfer methods simply focus on approximating the distribution of generated images with the distribution of the target domain. Inspired by the success of deep learning in multitasking [30, 31], we turn our attention to the image recognition task in the transferred domain that is free from image content exposure.

3. Domain Transferred Image Recognition

In order to transfer the secret image from the source domain to a target domain, an image generator is indispensable. Besides, a necessary classifier/recognizer will be responsible for image recognition in the transferred domain. As mentioned before, the domain transfer will alter the distribution of source images, in which way the privacy of images can be protected. However, changes in the distribution of data will hamper the image recognition task. Therefore, the generator

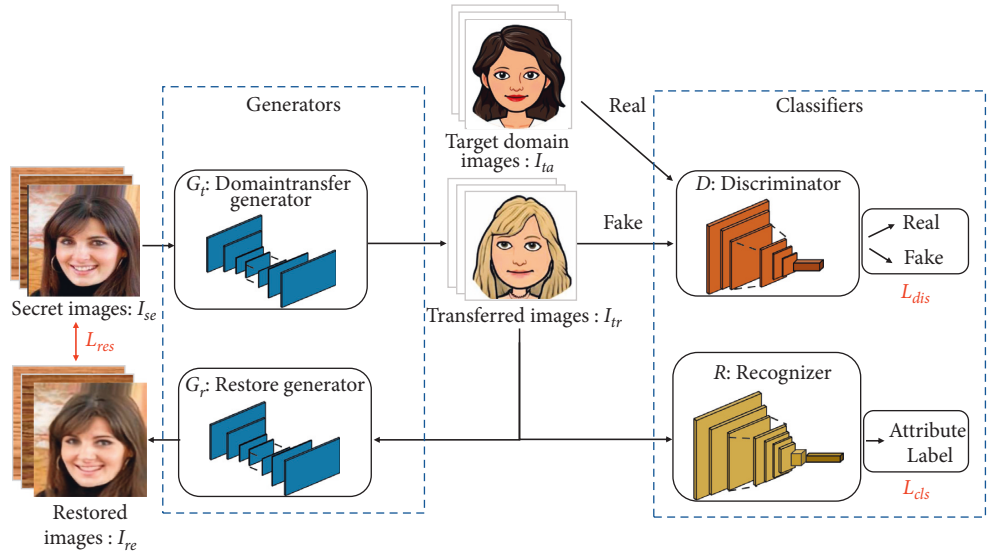

Figure 3: The schematic structure of our proposed system. For a secret image, it is firstly transferred to another image with a different style using the domain transfer generator. The transferred image serves as the input of the discriminator, recognizer, and restore generator. The recognizer can identify the attributes of the secret images on the transferred images. The restore generator is responsible for reconstructing the secret images based on transferred images. The discriminator aims to distinguish images in the target domain from images generated by the transfer generator.

and classifier modules will be trained in turn, just as the way of training a generative adversarial network. Besides, an alternative restore generator can also be incorporated to restore the secret image in some cases. Therefore, as shown in Figure 3, our whole framework mainly contains four modules: two generators, one discriminator, and one recognizer.

In order to describe the proposed method more clearly, we define the relevant concepts and variables as follows. The secret image dataset/domain is denoted as $\mathcal{X}_{\text{natural}}$, where the image in it is represented by I_{se}. Similarly, the target dataset/domain is defined as $\mathcal{X}_{\text{target}}$, where the image in it is denoted as I_{ta}. I_{tr} and I_{re} respectively indicate the transferred images output by the domain transfer generator and the restored images output by the restore generator. G_t and G_r represent the generator for the domain transfer and the generator for original secret image restoration, respectively. The discriminator is denoted by $\mathscr{D}$, and the recognizer $\mathscr{R}$ can identify secret image attributes on the transferred domain images I_{tr}.

3.1. Domain Transfer Generator. Given a secret image I_{se} from the dataset $\mathcal{X}_{\text{natural}}$ to be identified, the domain transfer generator (corresponding to G_t in Figure 3) aims to transfer the secret image into an image in the target domain with a different style such as a cartoon face style and an animal face style. In other words, the transfer generator G_t should extract the feature of the secret image as much as possible and transform this feature into an image that conforms to the distribution of the target domain. Besides, the transferred image must fool the discriminator to make it fail to tell whether this image is generated by G_t or comes from $\mathcal{X}_{\text{target}}$. When the whole system is fully trained, the generator G_t can be used as an independent module to perform image transfer from $\mathcal{X}_{\text{natural}}$ domain to $\mathcal{X}_{\text{target}}$ domain.

The domain transfer process can be written as

$$I_{tr} = G_t(I_{se}). \tag{1}$$

As for the design of our domain transfer generator, given that the convolutional neural network is experimentally proven to have excellent feature learning and extraction abilities for images and that a large amount of autoencoders have shown extraordinary performance in image generation, we adopt the combination of convolutional neural network and autoencoder and design the generator G_t with a ResNet [32] structure. Specifically, as shown in Figure 4, we first stack 3 sets of convolutional layers, instance normalization layers, and ReLU activation, followed by 9 residual blocks, with each residual block containing two convolutional layers, two instance normalization layers, and one ReLU activation. The input of each residual block and the output of the latter instance normalization will be added as the output of each residual block. Finally, 2 sets of deconvolution layers, ReLU activation layers, and instance normalization layers are followed to make the size of the final generated image consistent with that of the original image. By the way, the last activation layer will be tanh to cover the full pixel range.

3.2. Discriminator. For deep models based on generative adversarial networks, the discriminator $\mathscr{D}$ is indispensable in ensuring the quality of the generated images. The discriminator is expected to learn features that can distinguish the image I_{ta} in the target domain from the generated image I_{tr} in the transferred domain. When the input image is from the target domain $\mathcal{X}_{\text{target}}$, the discriminator should identify it as a "real" image, and when the input image is generated by

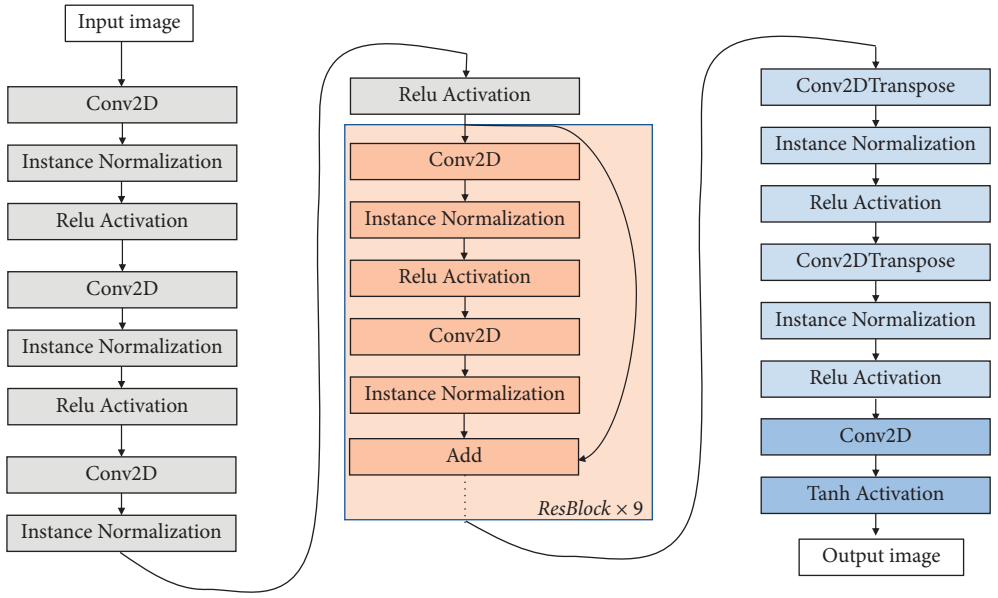

Figure 4: Structure of domain transfer generator G_t and restore generator G_r.

the generator G_t, the discriminator should identify it as a "fake" image, which can be mathematically expressed as

$$t = \mathscr{D}(I_{in}), \tag{2}$$

where t represents the output of discriminator $\mathscr{D}$, $t \in$ {"real": image from target domain, "fake": image generated by G_t}, and I_{in} represents the input image of $\mathscr{D}$.

As an auxiliary module of the generator G_t, for each input image, the discriminator $\mathscr{D}$ needs to judge whether it is "true" or "false." But, unlike generators which have to perform complex image generation tasks, the discriminator only makes a binary decision. Therefore, our framework only contains one discriminator, a very "shallow" network. The specific structure is shown in Figure 5(a). First, we stack a convolutional layer and a ReLU activation layer and then go through 3 convolutional blocks, each of which contains one convolutional layer, one instance normalization layer, and one ReLU activation layer, finally appending one convolutional layer as the end of the discriminator.

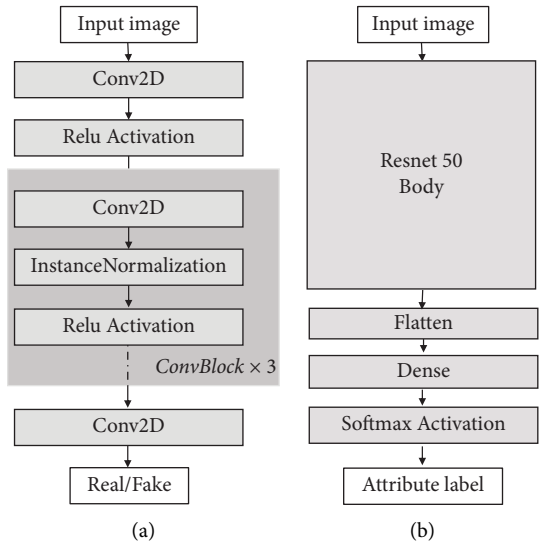

Figure 5: Structures of (a) discriminator and (b) recognizer.

3.3. Recognizer. For the recognition network $\mathscr{R}$, its main task is to be able to identify the attribute information of the original secret image I_{se} from the image I_{tr} generated by the domain transfer generator G_t, that is,

$$l = \mathscr{R}(I_{tr}), \tag{3}$$

where l is the predicted result.

Through the combination of several proposed modules, for a secret image I_{se}, the generator G_t firstly transforms the secret image to the target domain $\mathscr{X}_{\text{target}}$ and retains features that can characterize its attributes. Then, the recognizer $\mathscr{R}$ is able to extract this feature from the generated image and maps it to the attribute label representing the secret image. This process can be expressed as

$$\mathscr{R}: \left\{G_t: I_{se} \longrightarrow \mathscr{X}_{\text{target}}\right\} \longrightarrow l. \tag{4}$$

There are currently some popular network structures that perform very well in image recognition tasks, such as VGG [33] and ResNet [32]. Therefore, without loss of generality, we adopt ResNet structure as backbone for the recognizer in our framework and design different model heads according to the specific recognition task. Specifically, we take ResNet50 [32] as the backbone of our recognizer and replace the fully connected layer at the end of the model. The final number of output logits of the recognizer equals to the number of categories. The specific structure of the network is shown in Figure 5(b). After the input image passes through the backbone of ResNet50, the obtained features are flattened, passed through a fully connected layer, and activated by the

"softmax" function. Finally, the predicted label corresponds to the category label with the highest confidence.

3.4. Restore Generator. The recognizer designed in this paper can already recognize images, but in some scenarios that require more flexible authority certification or image processing operation, it may be necessary to recover the original secret image. Therefore, in addition to the recognizer for image recognition/authority certification, we also provide an additional restore generator G_t for image restoration. The restore generator receives the image generated by the domain transfer generator G_t as input and outputs an image as identical to the original secret image as possible. Since the task of the restore generator G_r is also to generate images, in order to reduce the complexity of the whole system, we set the structure of the restore generator G_r the same as the domain transfer generator G_t, which is shown in Figure 4.

3.5. Objective Loss Function. For a given secret image $I_{se} \in \mathcal{X}_{\text{natural}}$ and its label and an image in the target domain $I_{st} \in \mathcal{X}_{\text{target}}$, the designed network will be trained using the following adversarial loss.

For the discriminator $\mathcal{D}$, its input is the image I_{tr} generated by the generator or the image I_{ta} in the target domain $\mathcal{X}_{\text{target}}$, and its task is to be able to distinguish between these two kinds of images, which is a binary classification problem. In our framework, the images generated by the domain transfer generator are regarded as negative samples, and the images from the target domain are regarded as positive samples. Therefore, the discriminator will generate the following loss:

$$\mathcal{L}_{dis} = \mathbb{E}_{I_{st}}\left[\log \mathcal{D}\left(I_{st}\right)\right] + \mathbb{E}_{I_{se}}\left[\log\left(1 - \mathcal{D}\left(G\left(I_{se}\right)\right)\right)\right]. \quad (5)$$

For the recognizer $\mathcal{R}$, its purpose is to predict the attribute label of the secret image. Therefore, the cross-entropy loss function, most frequently used in image recognition, urges the recognizer to make correct prediction to the target label:

$$\mathcal{L}_{cls} = -\sum_{i=1}^{K} y_i \log\left(\mathcal{R}\left(I_{tr}\right)\right), \quad (6)$$

where K is the number of image categories, y is the image label, and $\mathcal{R}(I_{tr})$ is the predicted probability of the recognition model on the domain transferred image I_{tr}.

At the same time, the proposed framework also provides the function of restoring the original secret image. That is to say, the output of the restore generator G_r should be as same as possible to the secret image. To this end, the following loss function controls the similarity between I_{se} and I_{re}:

$$\mathcal{L}_{res} = \left\| I_{se} - I_{re} \right\|_2. \quad (7)$$

Finally, we will obtain a total loss as follows:

$$\mathcal{L}\left(G_r, G_t, G, \mathcal{R}\right) = \mathcal{L}_{dis} + \mathcal{L}_{cls} + \mathcal{L}_{res}. \quad (8)$$

4. Experiments

In this section, we will first introduce our datasets and evaluation metrics and experimental details. Then, we will demonstrate the effectiveness of our method on varied datasets.

4.1. Dataset. Since the proposed method is to transform images in one dataset/domain to another dataset/domain, for a complete domain transfer, two datasets are required, namely, the secret image dataset/domain $\mathcal{X}_{natural}$ and the target dataset/domain $\mathcal{X}_{\text{target}}$. We select face images indicating strong privacy as our source dataset/domain, and we adopt CelebA [34] and Pubfig [35] dataset in the experiment.

CelebA [34] is a large-scale face image database containing 202,599 images with 40 categories collected from the Internet by the Chinese University of Hong Kong. Each of these images has 40 binary attributes, such as gender, attractive or not, and young or not. Without loss of generality, we use the attribute "gender" as our prediction attribute, and all the images will be resized to resolution of 256×256. The first 10K images and the subsequent 2K images are respectively used as the training dataset and the test dataset in the experiment.

Pubfig [35] is a face image dataset with 58,797 images of 200 categories collected from the Internet. Each category has an average of 300 face images of one person. However, due to the copyright and privacy issue, most of the image links provided by the paper [35] are invalid now. As an alternative, we use the version published by other user on the Kaggle platform [36] including only 11,640 images with 150 categories. Specifically, we randomly choose 80% images as training dataset and the remain 20% images as testing dataset.

For the target dataset/domain $\mathcal{X}_{\text{target}}$, we mainly use Bitmoji-style cartoon face images. The Bitmoji [37] dataset is a cartoon style face downloaded directly from the mobile app. The Bitmoji [37] dataset contains 4085 faces with the resolution of 384×384. In the experiments, all the Bitmoji images are resized to the same size as the source domain $\mathcal{X}_{\text{natural}}$, with the resolution of 256×256. Figure 6 illustrates some examples of the Bitmoji dataset.

4.2. Evaluation. Since our framework is to perform image recognition in the transferred domain, the accuracy of image recognition is one of our goals. At the same time, we also provide a restore generator G_r to recover the original secret image. Therefore, the PSNR and SSIM metrics, which are most commonly used in digital image processing, are used to measure the quality of the recovered image. Given a reference image $\mathbf{I}$ and a test image $\mathbf{K}$, both with size $m \times n$, the PSNR between $\mathbf{I}$ and $\mathbf{K}$ is defined as

$$PSNR = 10 \cdot log_{10}\left(\frac{L^2}{MSE}\right), \quad (9)$$

where L is the dynamic range of allowable image pixel intensities (usually takes 255). For our 3-channel color image, we first calculate the MSE value of each channel and then calculate the average to get the MSE in equation (9).

Figure 6: Examples of Bitmoji images. The Bitmoji images contain cartoon faces of different genders, different hair colors, etc., all of which are centered and surrounded by white space.

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[\mathbf{I}(i,j) - \mathbf{K}(i,j)]^2. \tag{10}$$

The SSIM is given by

$$SSIM = \frac{(2\mu_{\mathbf{I}}\mu_{\mathbf{K}} + c_1)(2\sigma_{\mathrm{IK}} + c_2)}{(\mu_{\mathrm{I}}^2 + \mu_{\mathrm{K}}^2 + c_1)(\sigma_{\mathrm{I}}^2 + \sigma_{\mathrm{K}}^2 + c_1)}, \tag{11}$$

where $\mu_{\mathbf{I}}$ and $\mu_{\mathbf{K}}$ are the local means, σ_{I} and σ_{K} are the standard deviations and σ_{IK} is the cross-covariance for images $\mathbf{I}$ and $\mathbf{K}$ sequentially, and c_1 and c_2 are 6.50 and 58.5, respectively, by default.

4.3. Implementation Details. The implementation is based on Keras with TensorFlow as the backend. In our experiments, we use Adam [38] optimizer with a learning rate of 0.001 and linearly decay it to 0 after 50 training epochs. Our training batch size is set 4 and it takes our 4 days for training about 200 epochs on one single NVIDIA RTX 1080 Ti GPU.

4.4. Experimental Results. We first take the CelebA dataset as the domain $\mathcal{X}_{\text{natural}}$, the Bitmoji dataset as the domain $\mathcal{X}_{\text{target}}$, and the "gender" attribute in the CelebA dataset as the recognized attribute. After sufficient training, the obtained domain transferred images and the restored images are shown in Figure 7, where images in the first row belong to the CelebA dataset, images in the second row are transferred by the domain transfer generator G_t, and images in the last row are the recovered images. Visually, the transferred face image is similar to the cartoon face image in the Bitmoji dataset, which are all centered, frontal, and surrounded by white space. It is difficult to distinguish between these two kinds of images visually.

In order to verify the generalization of our method, we also use Pubfig dataset and the Bitmoji dataset as $\mathcal{X}_{\text{natural}}$ and $\mathcal{X}_{\text{target}}$ respectively for training, where the identity of Pubfig images is used as the prediction label. The results are shown in Figure 8.

4.4.1. Comparison of Other Methods. Table 1 shows the comparison of our method with other related works from various aspects. Tao et al. [39] and Baluja [25] focus on the secure secret message communication but failed in image recognition. The domain transfer methods [40–42] are free from carrier selection but none of them take secure image recognition into consideration. Our method can not only support secret image recovery and direct image recognition but also is free from carrier selection. Besides, the difference between our secret images and transferred images is enough (middle degree) to prevent the adversary from inferring the image content.

4.4.2. Visualization of Domain Transfer. Since the proposed model needs to transfer the image from one domain to another domain for "camouflage," the features of images obtained by domain transfer should be as close to the target domain $\mathcal{X}_{\text{target}}$ as possible. In the field of machine learning, there are some classic feature compression/dimension reduction methods, such as PCA [43], t-SNE [44], and LLE [45]. In order to explicitly portray the domain transfer process of our method, we visualize the feature distribution of the dataset $\mathcal{X}_{\text{natural}}$ before and after domain transfer and the feature distribution of target dataset $\mathcal{X}_{\text{target}}$ using t-SNE. Specifically, we first flatten all the CelebA and Bitmoji images from 256×256 to $1 \times 256 * 256 * 3$ and then directly apply the t-SNE function in the Sklearn [46] library to compress them into 1×2 and characterize these images on a 2D plane. The visualization of the dataset is shown in Figure 9. From Figure 9(a), we can see that the boundary between CelebA and Bitmoji is very obvious. But after transferring images in CelebA, the distribution of transferred CelebA and Bitmoji datasets is very similar, shown by the overlapping between the red dots and blue dots in Figure 9(b).

4.4.3. Recognition of Domain Transferred Image. In our proposed framework, after the original secret face image is transformed by the domain transfer generator G_t, the corresponding recognizer $\mathcal{R}$ should be able to directly identify the original secret image according to the transferred image. To examine the performance of the trained recognizer $\mathcal{R}$, we also train a recognition network directly on the original natural face as the best recognizer for comparison. Experiments are shown in Table 2. From the table, we can see that even if the image is transferred into other domains, our image recognition accuracy hardly decreases compared to the highest untransferred image recognition accuracy. For CelebA images, the recognition accuracy drops from 92.4% $\longrightarrow$ 92.1% and Pubfig images drop from 89.7% $\longrightarrow$ 88.4%.

4.4.4. Recognition of Reconstructed Images. As mentioned before, in order to use our proposed method more flexibly in some scenarios where the original secret image needs to be recovered, we also include a restore generator G_r for recovering the original secret image, and the usage of the restored image should not be affected. In order to test the effect of domain transfer on secret images, we test the image recognition performance of restored images. Specifically, we

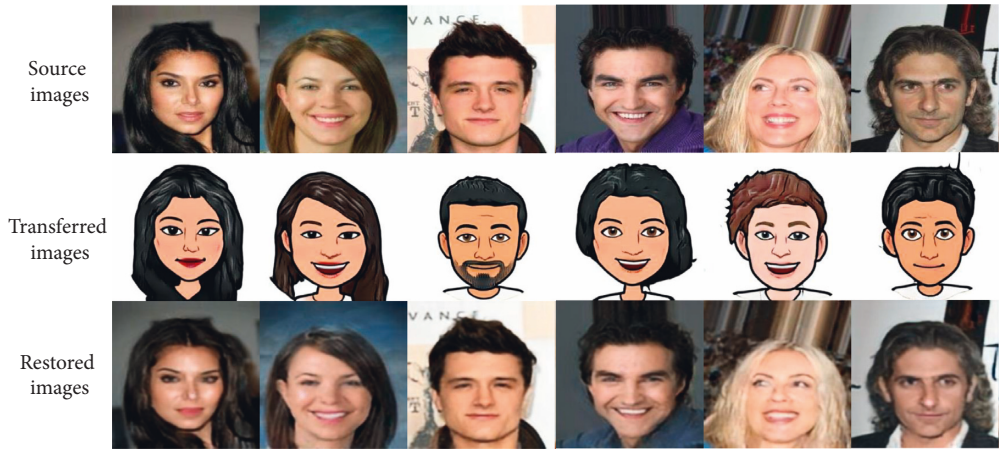

Figure 7: Visualization of domain transferred CelebA images and the corresponding restored images. From top to last row are original secret images I_{se}, domain transferred images I_{tr}, and recovered secret images I_{re}, respectively.

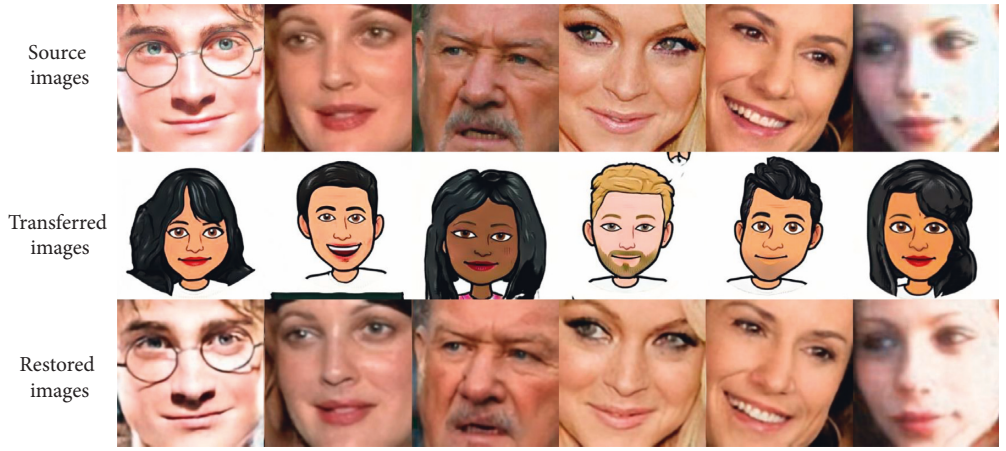

Figure 8: Visualization of domain transferred Pubfig images and the corresponding restored images. From top to last row are original secret images I_{se}, domain transferred images I_{tr}, and recovered secret images I_{re}, respectively.

Table 1: Comparison of relevant studies from different aspects. Difference means the difference between the secret image and the stego/generated/transferred image. STC means syndrome trellis coding; CNN and GAN mean convolutional neural network and generative adversarial networks, respectively.

Ref.	Carrier required	Difference	Technology adopted	Support secret images recovery	Support direct image recognition
Tao et al. [39]	Yes	Large	STC	Yes	No
Baluja [25]	Yes	Large	CNN	Yes	No
Kim et al. [40]	No	Middle	GAN	Yes	No
Chen et al. [41]	No	Middle	GAN	Yes	No
Liu et al. [42]	No	Small	GAN	No	No
Ours	No	Middle	GAN	Yes	Yes

train several recognizers on natural face images with different kinds of attributes and use these trained recognizers to test the image recognition accuracy on restored images. Without loss of generality, we select four attributes of "Male," "Bald," "Heavy_Makeup," and "Attractive" on the CelebA dataset to test the recognition performance of reconstructed images. The experimental results are shown in Table 3.

As can be seen from Table 3, when the network is trained directly with the original CelebA dataset, the recognition accuracies of "Male," "Bald," "Heavy_Makeup," and "Attractive" attributes are 92.4%, 98.2%, 90.0%, and 80.2%, respectively. Even if the original image has gone through the domain transfer, the recognition accuracy of the restored image is hardly affected, with the accuracy of "Male," "Bald," "Heavy_Makeup," and "Attractive" attributes being 92.2%, 98.2%, 87.7.0%, and 79.7%, respectively. Compared with the highest recognition accuracy of plaintext/baseline, the average value of the recognition accuracy has only dropped by less than one point (90.2% $\longrightarrow$ 89.5%), indicating that the

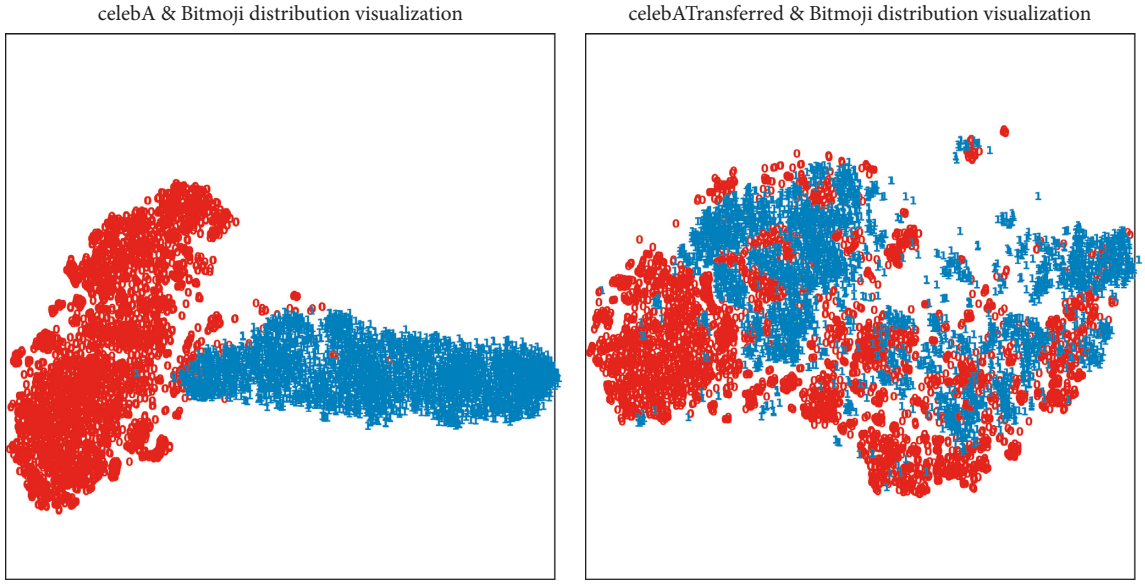

Figure 9: Visualization of the dataset after reducing the images to 2 dimensions using the t-SNE [44] algorithm. (a) The distribution of the CelebA dataset and Bitmoji dataset. The red dot means images in the CelebA dataset, and the blue dot means images in the Bitmoji dataset. (b) The distribution of the domain transferred CelebA dataset and Bitmoji dataset.

Table 2: Recognition accuracy comparison. The ACC in the "CelebA" column and "Pubfig" column means the top accuracy obtained by recognizers trained by natural, unaltered face images. The ACC in the "CelebA-T" column and "Pubfig-T" column means recognition accuracy obtained on transferred images.

Dataset	CelebA	CelebA-T	Pubfig	Pubfig-T
ACC (%)	92.4	92.1	89.7	88.4

Table 3: The recognition accuracy of CelebA and reconstructed CelebA images. ACC(Base) means the best accuracy obtained by the model trained on natural images. ACC(Res) means the recognition accuracy obtained on the restored images.

Attributes	Male	Bald	Heavy_Makeup	Attractive	Average
ACC(Base)	92.4	98.2	90.0	80.2	90.2
ACC(Res)	92.2	98.2	87.7	79.7	89.5

image reconstructed in our method can still be used with little performance penalty.

4.4.5. Visual Quality of Reconstructed Images. From the perspective of secret information transfer, the process of recovering the original secret image in our method can also be used for secret communication. The method of hiding images within images mentioned in [25] is an advanced method based on neural networks for large capacity information hiding, where the author pointed out that if the natural image is directly fed into the proposed neural network, the hidden image will be exposed in the residual image. To eliminate the traces of hidden image content in residual images between cover images and stego images, a simple way is to permute the pixels of hidden images before they are passed to the preparation network. Following [25], we retrained our whole network to hide images without the spatial coherence of natural images. As shown in Table 4, although the average values of PSNR and SSIM of the restored images are slightly inferior to the shuffled version, the recognition of restored images will be little affected. One thing that needs to noted is that we are only sacrificing a little bit of image quality to enable image recognition while preserving privacy. Besides, our method is free from cover image selection hence decreasing the complexity of the system during the usage/inference phase.

Table 4: Quantitative visual quality of recovered images.

Method	Baluja [25]	Baluja (shuffled) [25]	Ours
PSNR	34.2	31.6	26.4
SSIM	0.962	0.943	0.948

5. Conclusion

In this paper, we proposed a technique for image recognition while protecting the privacy of image content. First, we point out that our method is free from not only complex computation such as encryption algorithms but also carrier selection like information hiding-based method. Second, we designed and trained a combined network containing two generators, one recognizer, and one discriminator, where these two generators are responsible for domain transfer and image reconstruction, respectively, and the recognizer for image recognition on domain transferred images. Experiments are conducted on several standard datasets and the results have validated the effectiveness of our proposed method.

Data Availability

The CelebA data used to support the findings of this study are from previously reported studies and datasets, which have been cited. The Pubfig and Bitmoji data used to support the findings of this study were supplied by Kaggle under license, which can be downloaded by hyperlinks provided in references [36] and [37], respectively.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] P. Li, J. Li, Z. Huang et al., "Multi-key privacy-preserving deep learning in cloud computing," *Future Generation Computer Systems*, vol. 74, pp. 76–85, 2017.

[2] X. Liu, L. Xie, Y. Wang et al., "Privacy and security issues in deep learning: a survey," *IEEE Access*, vol. 9, pp. 4566–4593, 2020.

[3] Y. Qu, S. Yu, L. Gao, W. Zhou, and S. Peng, "A hybrid privacy protection scheme in cyber-physical social networks," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 3, pp. 773–784, 2018.

[4] Y. I. Daradkeh, I. Tvoroshenko, V. Gorokhovatskyi, L. A. Latiff, and N. Ahmad, "Development of effective methods for structural image recognition using the principles of data granulation and apparatus of fuzzy logic," *IEEE Access*, vol. 9, Article ID 13417, 2021.

[5] C. Luo, L. Jin, and Z. S. Moran, "A multi- object rectified attention network for scene text recognition," *Pattern Recognition*, vol. 90, pp. 109–118, 2019.

[6] A. B. Nassif, S. Ismail, I. Attili, M. Azzeh, and K. Shaalan, "Speech recognition using deep neural networks: a systematic review," *IEEE Access*, vol. 7, Article ID 19143, 2019.

[7] M. Xue, C. Yuan, H. Wu, Y. Zhang, and W. Liu, "Machine learning security: threats, countermeasures, and evaluations," *IEEE Access*, vol. 8, Article ID 74720, 2020.

[8] J. Han, W. Zang, Y. Meng, and R. Sandhu, "Quan- tify co-residency risks in the cloud through deep learning," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 4, pp. 1568–1579, 2020.

[9] O.-A. Kwabena, Z. Qin, T. Zhuang, and Z. Qin, "MSCryptoNet: multi-scheme privacy-preserving deep learning in cloud computing," *IEEE Access*, vol. 7, Article ID 29344, 2019.

[10] Z. Li, W. Xu, H. Shi, Y. Zhang, and Y. Yan, *Security and Privacy Risk Assessment of Energy Big Data in Cloud Environment*, Computational Intelligence and Neu- roscience, vol. 2021, , Article ID 2398460, 11 pages, 2021.

[11] Y. Liu, Z. Ma, X. Liu, S. Ma, and K. Ren, "Privacy-preserving object detection for medical images with faster rcnn," *IEEE Transactions on Information Forensics and Security*, vol. 17, 2019.

[12] X. Liao, Y. Yu, B. Li, Z. Li, and Q. Zheng, "A new payload partition strategy in color image steganography," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 3, pp. 685–696, 2019.

[13] H. Zhao, Q. Dai, J. C. Ren, W. Wei, Y. Xiao, and C. Li, "Robust information hiding in low-resolution videos with quantization index modulation in dct-cs domain," *Multimedia Tools and Applications*, vol. 77, no. 14, Article ID 18827, 2018.

[14] G. Craig, "Fully homomorphic encryption using ideal lattices," in *Proceedings of the 40th-first annual ACM symposium on Theory of computing*, pp. 169–178, May 2009.

[15] A. Sanyal, M. J. Kusner, A. Gasco´n, and V. Kanade, "Tapas: tricks to accelerate (encrypted) prediction as a service," in *Proceedings of the 35th International Conference on Machine Learning*, vol. 80, pp. 4490–4499, 2018.

[16] B. Knott, S. Venkataraman, A. Hannun, S. Sengupta, M. Ibrahim, and L. van der Maaten, "Crypten: secure multi-party computation meets machine learning," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[17] A.-T. Tran, T.-D. Luong, J. Karnjana, and V.-N. Huynh, "An efficient approach for privacy preserving decentralized deep learning models based on secure multi- party computation," *Neurocomputing*, vol. 422, pp. 245–262, 2021.

[18] D. Anders, D. Escudero, and M. Keller, "Secure evaluation of quantized neural networks," *Proceedings on Privacy Enhancing Technologies*, vol. 2020, no. 4, pp. 355–375, 2020.

[19] M. S. Riazi, M. Samragh, H. Chen, L. Kim, K. Lauter, and F. Koushanfar, "Xonn: xnor- based oblivious deep neural network inference," in *Proceedings of the 28th USENIX Conference on Security Symposium*, pp. 1501–1518, August 2019.

[20] S. Dhawan and R. Gupta, "Analysis of various data security techniques of steganography: a survey," *Information Security Journal: A Global Perspective*, vol. 30, no. 2, pp. 63–87, 2021.

[21] W. Shen, J. Qin, Y. Jia, H. Rong, and J. Hu, "Enabling identity-based integrity auditing and data sharing with sensitive information hiding for secure cloud storage," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 2, pp. 331–346, 2018.

[22] Y. Yang, Z. Li, W. Xie, and Z. Zhang, "High capacity and multilevel information hiding algorithm based on pu partition modes for hevc videos," *Multimedia Tools and Applications*, vol. 78, no. 7, pp. 8423–8446, 2019.

[23] Y. Uchida, Y. Nagai, S. Sakazawa, and S. Satoh, "Embedding watermarks into deep neural networks," in *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, pp. 269–277, June 2017.

[24] B. D. Rouhani, H. Chen, and F. Koushanfar, "Deepsigns: a generic watermarking framework for ip pro- tection of deep learning models," 2018, https://arxiv.org/abs/1804.00750.

[25] S. Baluja, "Hiding images within images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 7, pp. 1685–1697, 2019.

[26] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversar- ial networks," in *Proceedings of the IEEE conference on com- puter vision and pattern recognition*, pp. 126–137, ACM, Honolulu, HI, USA, July 2019.

[27] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle- consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, IEEE, Venice, October 2017.

[28] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: unified genera- tive adversarial networks for multi-domain image-to-image translation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8789–8797, IEEE, Salt Lake City, UT, USA, June 2018.

[29] H. Tang, H. Liu, and N. Sebe, "Unified generative adversarial networks for controllable image-to-image translation," *IEEE Transactions on Image Processing*, vol. 29, pp. 8916–8929, 2020.

[30] F. Liang, L. Zhou, J. Zhong et al., "Evolu- tionary multitasking via explicit autoencoding," *IEEE Trans- Actions on Cybernetics*, vol. 49, no. 9, pp. 3457–3470, 2018.
[31] S. Liu, Q. Shi, and L. Zhang, "Few-shot hy- perspectral image classification with unknown classes using multitask deep learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 6, pp. 5085–5102, 2020.
[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, Las Vegas, NV, USA, June 2016.
[33] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, https://arxiv.org/abs/1409.1556.
[34] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3730–3738, Santiago, Chile, December 2015.
[35] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and simile classifiers for face verification," in *Proceedings of the IEEE 12th international conference on computer vision*, pp. 365–372, IEEE, Kyoto, Japan, September 2009.
[36] K. Chaudhari, "Pubfig. Website," 2021, https://www.kaggle.com/datasets/kaustubhchaudhari/pubfig-dataset-256x256-jpg.
[37] M. Mozafari, "Bitmoji faces. Website," 2021, https://www.kaggle.com/mostafamozafari/bitmoji-faces.
[38] D. P. Kingma and B. A. Jimmy, "Adam: a method for stochastic optimization," 2014, https://arxiv.org/abs/1412.6980.
[39] J. Tao, S. Li, X. Zhang, and Z. Wang, "Towards robust image steganography," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 2, pp. 594–600, 2018.
[40] J. Kim, M. Kim, H. Kang, and K. H. Lee, "U-gat-it: unsupervised generative attentional net- works with adaptive layer-instance normalization for image- to-image translation," in *Proceedings of the International Conference on Learning Representations*, April 2019.
[41] R. Chen, W. Huang, B. Huang, F. Sun, and B. Fang, "Reusing discriminators for encoding: towards unsupervised image-to-image translation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8168–8177, Seattle, WA, USA, June 2020.
[42] M. Liu, Q. Li, Z. Qin, G. Zhang, P. Wan, and Z. Wen, "Blendgan: implicitly gan blending for arbitrary stylized face generation," *Advances in Neural Information Processing Systems*, vol. 34, 2021.
[43] S. Wold, E. Kim, and G. Paul, "Principal component analysis," *Chemometrics and Intelligent Labora- Tory Systems*, vol. 2, no. 1-3, pp. 37–52, 1987.
[44] L. Van der Maaten and G. Hinton, "Visualiz- ing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. 11, 2008.
[45] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
[46] Scikit, "Scikit-learn library. Website," 2021, https://scikit-learn.org/stable/.