Hindawi
Security and Communication Networks
Volume 2022, Article ID 3493336, 14 pages
https://doi.org/10.1155/2022/3493336

Research Article

An Improved Self-Training Model with Fine-Tuning Teacher/Student Exchange Strategy to Detect Computer-Generated Images

Ye Yao, Shuhui Liu, Hui Wang, Zhangyi Shen, and Xuan Ni

School of Cyberspace, Hangzhou Dianzi University, Hangzhou 310000, China

Correspondence should be addressed to Hui Wang; wanghui.ac@hotmail.com

Received 27 January 2022; Revised 25 February 2022; Accepted 11 March 2022; Published 16 April 2022

Academic Editor: Beijing Chen

Computer-generated (CG) images have become indistinguishable from natural images due to powerful image rendering technology. Fake CG images have brought huge troubles to news media, judicial forensics, and other fields. How to detect CG image has become a key point to solve the problems mentioned above. The image classification method based on deep learning, due to its strong self-learning ability, can automatically determine the differences in the image features between CG images and natural images and can be used to detect CG images. However, deep learning often requires a large amount of labeled data, which is usually a tedious and complex task. This paper proposes an improved self-training strategy with *fine-tuning teacher/student exchange* (FTTSE) to solve the problem of missing labeled datasets. Our method is actually a strategy based on semisupervised learning to train the teacher model through labeled data and to predict the unlabeled data by the teacher model to generate pseudo labels. The student model is obtained by continuous training on the mixed dataset composed of labeled and pseudo-labeled data. A teacher/student exchange strategy is designed for iterative training; i.e., the identities of the teacher model and the student model are exchanged at the beginning of each round of iteration. And then the new teacher model is used to predict pseudo labels, and the new student model exchanged from teacher model in the previous round of iteration is fine-tuned and retrained by the mixed dataset with new pseudo labels. Furthermore, we introduced malicious image attacks to perturb the mixed dataset to improve the robustness of the student model. The experimental results show that the improved self-training model we proposed can stably maintain the image classification ability even if the testing images are maliciously attacked. After iterative training, the CG image detection accuracy of the final model increases by 5.18%. The robustness against 100% malicious attacks is also improved, where the final trained model has an accuracy improvement of 7.63% higher than the initial model. The self-training model with FTTSE strategy proposed in this paper can effectively enhance the detection ability of the existing model and can greatly improve the robustness of the model with iterative training.

1. Introduction

With the advancement of computer image rendering technology, a computer-generated (CG) image has become an important visual information carrier. Because of their unique artistry and strong sense of reality, CG images have been widely used in people's daily life and entertainment, e.g., games, virtual reality, and 3D animation. With the advancement of powerful hardware-supported rendering technology and generative adversarial network (GAN) technology, the generation for CG images has been greatly simplified, and the generated image has become more and more realistic. It is hard to distinguish the CG images from natural images by using both human perception and computer detection. This means there are opportunities for malicious attackers to deceive facial recognition systems to impersonate others by using CG images and to create fake news to gain illegal profits, damage others' reputations, or maliciously create chaos. All the projects, such as the Digital Emily Project in 2010 [1], the Face2Face Project in 2016 [2], and the Synthesizing Obama Project in 2017 [3], prove that performing a spoofing attack has been greatly simplified, and the illegible CG images have created a focus of security concerns in the fields of news media and judiciary.

In order to solve the above mentioned problems, there are two kinds of methods proposed for CG images and natural images classification. One is based on handcrafted features [4–10] and the other is based on convolutional neural network (CNN) [11–15]. The former usually uses feature extractors for classification. The statistical properties can be obtained from transformed images by wavelet transform or other differential operators. Now the methods based on manual statistical feature has been widely used to distinguish CG images from natural images. Rahmouni et al. [11] prove the statistical-feature-based methods perform well in image classification. CNN-based methods map the images to their corresponding labels with a function to distinguish the CG images and natural images in an end-to-end manner. Compared with the methods based on manual statistical feature, CNN-based methods have strong learning ability and can automatically learn image features. Therefore, CNN-based methods usually achieve better performance and can significantly improve the accuracy of classification. The current state-of-the-art CG image detection models are trained based on supervised learning. However, a large amount of image labels are required for training these models, which cost a lot of time for collecting correct labels. If the unlabeled images can either be used for training, the problem of insufficient labels can be effectively solved.

Current state-of-the-art CG image detection models are not as robust to changes in distribution as humans. How to quickly adapt to such changes with few labeled examples for learning is the central issue to study. As proposed by Zhu et al. [16], domain adaption is a focused research trend for transfer learning to deal with the problem. The self-training strategy proposed by Xie et al. [13] also belongs to transfer learning methods. In [16], their main contribution is to improve the adversarial robustness by using unlabeled data, and the experimental result achieves a higher accuracy with a smaller ratio between labeled and unlabeled batch size (1 : 14 and 1 : 28). The self-training strategy proposed in [13] is based on semisupervised learning (SSL). The basic idea is to find a way to use unlabeled datasets to expand labeled datasets. They firstly train the teacher model using the standard cross entropy loss with labeled data. Then the pseudo labels are generated for unlabeled images by the teacher model. The equal-or-larger student model is trained with the combined data (labeled and pseudo-labeled data) and injected noise. The student model then is used as a new teacher model for iterative training until fitting. In this self-training strategy, the model trained based on labeled data is essentially the teacher model, and the model trained based on mixed data is essentially the student model. However, there are two main drawbacks of the self-training strategy proposed in [13]. Firstly, the teacher model is directly discarded after generating the pseudo labels, which not only wastes the computing resources but also wastes a lot of internal prior knowledge of prior model training. Secondly, the student model is directly trained with mixed labels, which results in the inability to improve the model robustness.

In order to overcome the drawbacks mentioned above, we propose an improved self-training strategy with *fine-tuning teacher/student exchange* (FTTSE) to distinguish CG images from natural images. Our contributions are summarized as follows.

(1) The FTTSE strategy is designed by fine-tuning a previous round of the teacher model to train the subsequent student model. The main difference between our proposed strategy and [13] is the teacher/student exchange strategy. The teacher model is not directly discarded after generating the pseudo labels in our work. Except for the initial-trained teacher and student models before iteration, the previous round of teacher model is fine-tuned with learning rate decay and then it exchanges the teacher/student identity for subsequent student model training. The subsequent student models are retrained on basis of the prior knowledge learned by the previous round of teacher model.

(2) A "Local-to-Global" strategy for pseudo-label construction is designed to reduce the interference of noisy labels in student model training. The improved pseudo-label construction strategy has strong flexibility for making classification decision no matter if by one-time prediction of the whole image or multiple-times predictions of the image blocks.

(3) Malicious attacks are added to the mixed labeled image dataset for training student model to enhance its robustness.

(4) A learning rate decay strategy is applied in FTTSE to prevent the model from falling into a local optimum.

The rest of this paper is organized as follows. Section 2 discusses existing related works, including the methods based on handcrafted feature, deep learning, and SSL. Section 3 illustrates the details of our proposed improved self-training strategy with FTTSE for CG image classification. In Section 4, we design a series of test experiments to verify the improved capability of the proposed strategy. This is followed by conclusion and future work in Section 5.

2. Related Works

There are two main categories of methods currently used to distinguish CG images from natural images. One is the method based on handcrafted features, which usually requires features extracted from spatial and transformed domain and uses support vector machine (SVM) as training classifier. The other is the method based on deep learning. For the handcrafted-feature-based method, Li et al. [17] proposed a multiresolution method to distinguish CG images and natural images by directly using the local binary pattern features of the image and the SVM classifier. Ng et al. [4] proposed a model based on physics-motivated features to assist in the recognition of CG images by statistically analyzing the differences between the image generation process in three aspects: object model difference, light transport difference, and image acquisition difference. In the paper, the physical characteristics, including the gamma correction in natural images and sharp structures in CG images, are described by means of the fractal geometry at the finest scale and the differential geometry at the intermediate scale. The geometry-based SVM classifier achieved good performance in terms of both speed and accuracy. Wu et al. [18] explored the image histogram directly as the main feature for classification. The several highest bins of different image histograms are extracted as classification features to identify CG images. Although the histogram features are simple, the classifier works well in terms of detection accuracy. Lyu and Farid [19] proposed a statistical model based on the features extracted from first-order and higher-order wavelet statistics to distinguish CG images from natural images. However, the design of handcrafted features is often complex and has to be self-created for making fine distinctions. These methods generally perform well on simple dataset with images collected from limited sources, whereas they often show performance limitations when the training process is encountered with a complex dataset with images collected from many sources. To consider both global visual features and finer differences for CG image forensic, Bai et al. [20] contributed a large-scale CG images benchmark (LSCGB) with large-scale images which contain CG images with four different scenes generated by various rendering techniques and are collected with small bias on the distributions of color, brightness, tone, and saturation. They also proposed an effective texture-aware network based on the texture difference between CG and natural images to improve forensic accuracy and to exhibit the feasibility of LSCGB.

Besides the handcrafted-feature-based method, the deep-learning-based method, especially the CNN-based method, has also become a popular classification technology and has been researched with more outstanding achievements. CNN-based model is usually in an end-to-end manner, which automatically learns appropriate feature representations from superficial layer to deeper layer by existing data information. Compared with the traditional methods based on handcrafted feature which are designed and extracted from prior knowledge and assumptions, the CNN-based methods are more suitable for classification of images collected from complex source scenarios because of the powerful self-learning ability for abstraction of data features. In our work, we propose an improved teacher/student self-training strategy to detect CG images, which is essentially a semisupervised and deep-learning method. In the following subsections, the research works for image classification will be further studied in aspects of deep learning and SSL strategies, respectively.

2.1. Methods Based on Deep Learning. Benefiting from the powerful learning ability of deep learning neural network, there are many methods based on deep learning proposed to solve the problem of CG image detection. Gando et al. [21] proposed a deep convolutional neural network (DCNN) model based on fine-tuning, which includes a custom CNN-based model trained from scratch and a traditional model using handcrafted features. The fine-tuned DCNN model can automatically distinguish aggregating illustrations from photographs with detection accuracy of 96.8%. Rahmouni et al. [11] designed a special pooling layer to extract feature statistics from complex images, which optimized the "end-to-end" CNN framework and enhanced the performance of distinguishing CG images and natural images. Yao et al. [22] proposed a CG image detection method based on sensor pattern noise and deep learning. In [22], the input images were filtered by three high-pass filters to remove low-frequency information, so as to eliminate the interference to the recognition accuracy. He et al. [23] combined CNN and recurrent neural network (RNN) to classify CG images and natural images. The authors design a dual-path neural network architecture using preprocessing operations of color space transformation and Schmid filter bank to extract image color and texture features. Exploiting the image color and texture features, the joint feature representations of local patches are learned to extract global artifact through a directed acyclic graph RNN. Capsule network was proposed in [24] and was extended in [25] to detect forged image and video for capsule-forensics applications. Tarianga et al. [26] proposed a deep convolutional recurrent model based on efficient attention.

Recently, Zhang et al. [27] used channel and pixel correlation information to reveal different features between CG images and natural images. In [27], a self-coding module was designed at the beginning of CNN and was utilized to deeply explore the correlation between the three color image channels. A new end-to-end CNN architecture called self-coding network (ScNet) was constructed with introducing hybrid correlation module and combining with existing CNN model to enhance the discrimination ability and application generality. Quan et al. [28] pointed out that the problem of blind detection of CG images is ignored in existing CNN-based methods, i.e., it is unknown whether the training images is generated by computer rendering tools or not for detection training. In order to improve the generalization ability of the model, a dual-branch neural network was designed to capture diverse features. After the normal training, the gradient information based on the CNN model is used to generate harder negative samples and then conduct enhanced training using both the original training samples and the generated negative samples. Huang et al. [29] proposed a method for effectively identifying three different kinds of digital image origin based on CNN and used a local-to-global framework to reduce training complexity. In their work, the raw pixels are used as input to CNN without the aid of "residual map." The method behaves robustly against several common postprocessing operations.

In the latest research study, Meena and Tyagi [30] proposed a two-stream convolutional neural network to distinguish CG and photographic images. The first stream branch in the network focuses on learning different features of RGB images, while the second stream branch focuses on learning the noise features. Finally, the outcomes of two streams are merged using ensemble learning model. The experimental results show that the method maintains good performance even if the test image is processed with Gaussian noise. There are still some research works to expand the forensic functions in some special CG image application scenarios. In [31], the deep neural network was applied for image copy-move forgery localization. In [32], an improved Xception model was applied for realistic fake face images detection. The fake face images are generated by GAN, and the experimental results show a detection accuracy improvement with the designed robust dual-stream network.

In this paper, we focus on the image classification in the application scenario of distinguishing CG images and photographic images. The network of ScNet proposed by Zhang et al. [27] adopts a distinctive mixed-channel correlation module and has strong discriminative ability and generality. Therefore, we choose ScNet as the base model for our experiments to verify the performance improvement of our proposed self-training model with FTTSE strategy.

2.2. Methods Based on SSL. Deep neural network is the state-of-the-art technology in various image classification applications, which has also obtained remarkable achievement in the research field of distinguishing CG images from natural images. However, the challenge in this research field is how to overcome the lack of labeled data to train the complex network. It is an expensive and time-consuming work to obtain a large amount of labeled data for different image classification tasks. Meanwhile, it is not feasible to manually label the data on a large scale because of the data privacy and access restrictions.

The network based on SSL proposed in [33] is one of the effective methods to solve the above problems by utilizing some labeled data and a large amount of unlabeled data for training. Mukherjee and Awadallah [34] propose an improved self-training approach that combines Bayesian deep learning and uncertainty estimation from the underlying neural network. Generally speaking, this method is a learning mechanism that uses Monte Carlo dropout as an acquisition function, selects instances from an unlabeled data pool, and uses model confidence for self-training. This method achieves excellent performance on large-scale pre-trained language models. Xie et al. [13] proposed a self-training model based on standard SSL strategy. This method firstly trains an efficient network model as teacher model on labeled images and generates pseudo labels on unlabeled images. Then it trains a higher efficient network as student model on a collection of labeled and pseudo-labeled images. The classification ability of student model is enhanced through iterative training by putting itself into the teacher model position. Compared with other SSL based models, this method improves the detection accuracy by 2.0% for ImageNet classification. Zou et al. [35] considered the noisy problem caused by the predicted pseudo labels that may result in overconfidence and wrongly placing labels on their classification in the process of self-training. In iterative training, this error bias may also propagate with iterations. To solve the problem, a self-training framework with confidence regularization was proposed. Chen et al. [36] studied an SSL model in a class-imbalanced data environment, using SSL for generating high-precision pseudo labels on minority classes. In [36], a class rebalance self-training (CReST) framework was proposed to improve existing SSL-based methods for handling class-imbalanced data. The framework iteratively retrains the baseline SSL model to expand the sample labels by adding sample pseudo labels from the unlabeled dataset, where pseudo-labeled samples from the minority class are selected according to the estimated class distribution. They also proposed a new distribution alignment to adaptively adjust the rebalance strength, which had an outperformance comparing with other rebalancing methods based on SSL.

3. The Proposed Algorithm for CG Image Detection

A self-training strategy proposed by Xie et al. [13] is a method based on SSL to augment labeled datasets with unlabeled datasets. But in this method, the trained teacher model is only used to generate pseudo labels in each round of iteration and then is discarded in the next round of iteration. This results in abandoning a large amount of prior knowledge learned within the teacher model. And the computing resources consumed for training the teacher model are also wasted. In this paper, we propose an improved teacher/student exchange self-training strategy with FTTSE. The design process of the algorithm is presented in Algorithm 1. In our improved strategy, all the subsequent models are fine-tuned on the basis of the model trained in the previous round of iteration except the initial-trained teacher and student models. We design a pseudo-label construction strategy for predicting unknown image samples. In order to improve the robustness of our model, malicious attacks are applied to the images in the mixed dataset. A learning rate decay strategy is also adopted to speed up the model fitting while avoiding model falling into local optimum. Figure 1 presents the flowchart of the improved self-training model with FTTSE strategy for CG image detection.

3.1. Strategy of Teacher/Student Exchange. In this paper, we have improved the self-training strategy with teacher/student iterative training proposed by Xie et al. [13]. In the following, our proposed self-training model with FTTSE strategy will be introduced with more design details.

Before teacher/student exchange iteration, the initial teacher model θ_0 is trained from labeled data D. The teacher model θ_0 is used to predict the pseudo label $\tilde{y}_i$ on unlabeled dataset $\tilde{D}$ and then combine the pseudo-labeled dataset and original labeled dataset together to generate a mixed dataset D'. The student model θ_1 is trained on D_a' which is generated from D' with adding malicious attacks. Until now, the initial teacher model θ_0 and initial student model θ_1 are obtained by the first TWICE training.

Then the teacher/student exchange iteration begins. Take the first round of iteration as example. The teacher and student models firstly exchanged their identities; that is, (1) θ_1 will be used as the teacher model to generate new pseudo-label dataset; (2) θ_0 will be regarded as student model and fine-tuned for training a new student model θ_2 with new

```
Input: labeled dataset $D = \{(x_i, y_i)\}_{i=1}^{n}$ & unlabeled dataset $\widetilde{D} = \{(\widetilde{x}_i)\}_{i=1}^{t}$
Output: model $\theta_k$
Train teacher model $\theta_0$ with labeled dataset $D$.
    $(1/n)\sum_{i=1}^{n} l(y_i, f(x_i, \theta_0))$, $\forall i = 1, 2, \ldots, n$
$(1/n)\sum_{i=1}^{n} l(y_i, f(x_i, \theta_0))$, $\forall i = 1, 2, \ldots, n$Use the trained teacher model $\theta_0$ to predict the pseudo label $\widetilde{y}_i$ on unlabeled dataset $\widetilde{D}$.
    $\widetilde{y}_i = P(\widetilde{x}_i, \theta)$, $\forall i = 1, 2, \ldots, t$
    $\widetilde{y}_i = P(\widetilde{x}_i, \theta)$, $\forall i = 1, 2, \ldots, t$And then combine the pseudo-labeled dataset $D' = \{(\widetilde{x}_i, \widetilde{y}_i)\}_{i=1}^{t}$ and the labeled dataset $D$ together to generate a mixed dataset. $D' = \{(x_i, y_i,)\}_{i=1}^{n+t}$
Apply malicious attacks to $D'$ to get attacked mixed image dataset $D'_a = \{(x_i^a, y_i^a,)\}_{i=1}^{n+t}$
Train the student model $\theta_1$ with the attacked mixed image dataset $D'_a$.
    $(1/n+t)\sum_{i=1}^{n+t} l(\widetilde{y}_i^a, f(\widetilde{x}_i^a, \theta_1))$, $\forall i = 1, 2, \ldots, n+t$
$1/n+t\sum_{i=1}^{n+t} l(\widetilde{y}_i^a, f(\widetilde{x}_i^a, \theta_1)), k = 1$
    $\forall i = 1, 2, \ldots, n+t$.
while neural network unfitting do
    Last student model $\theta_k$ is regarded as the $k^{\text{th}}$ teacher model to predict pseudo label $\widetilde{y}_i$ and generate a new attacked mixed dataset $D'_a$
        $\theta_k \longrightarrow^{\text{predict}} \widetilde{D} \longrightarrow^{\text{combine}} D' \longrightarrow^{\text{attack}} D'_a$
    $\theta_k \longrightarrow^{\text{predict}} \widetilde{D} \longrightarrow^{\text{combine}} D' \longrightarrow^{\text{attack}} D'_a$.   Last teacher model trained from previous round of iteration: $\theta_{k-1}$ is fine-tuned and used for retraining student model $\theta_{k+1}$ with dataset $D'_a$.
        $\theta_{k+1} = f(\theta_{k-1}, D'_a)$
    $\theta_{k+1} = f(\theta_{k-1}, D'_a)$.   Implement the learning rate decay strategy to reduce the learning rate value as follows:
        $lr_{k+1} = (1/10) lr_k$
    $lr_{k+1} = (1/10) lr_k . k++$
end while
```

Algorithm 1: Improved self-training algorithm with FTTSE.

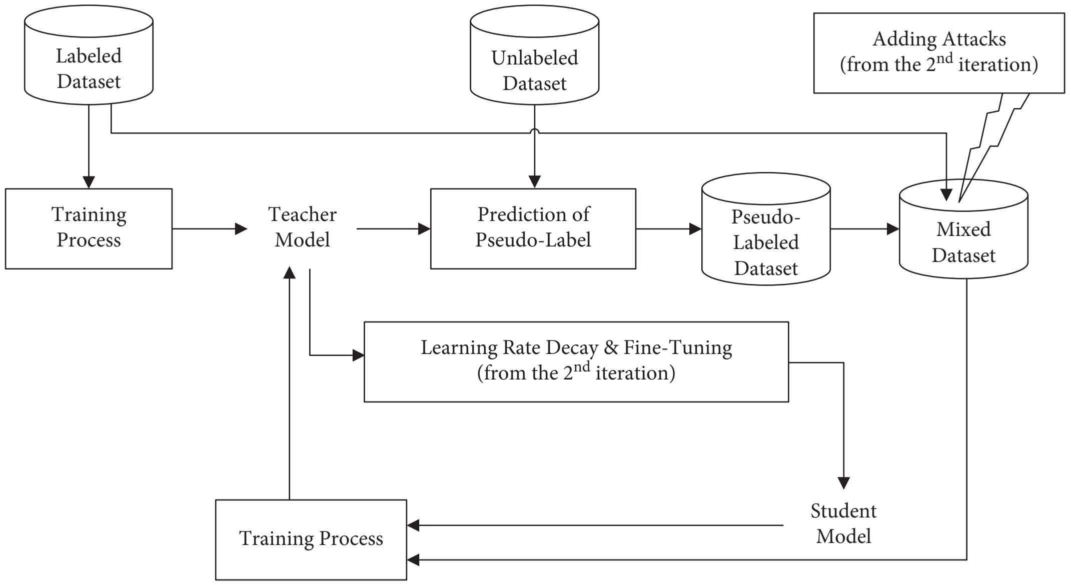

Figure 1: Improved self-training model with FTTSE for CG image detection.

attacked mixed dataset. At the end of each round of iteration, the learning rate will be reduced by our learning rate decay strategy. As the model θ_k continues to self-train in backward k^{th} round of iteration, the identities of teachers and students are exchanged for pseudo-label prediction and model retraining over and over again.

The improved teacher/student exchange strategy can retain the weights of the teacher model trained from the previous round of iteration and convert them into a student model with fine-tuning to learn new image contents and distribution features. The main improvement of the proposed teacher/student exchange strategy is reflected in the generalization ability of the model. Our approach actually draws on the idea of transfer learning, which is helpful to avoid wasting of computing resources and prior knowledge and greatly speed up model training.

3.2. Strategy of Pseudo-Label Construction. In our proposed method, the pseudo label predicted by the teacher model is one kind of hard labels that conforms to a one-hot distribution. For an image x, the predicted value with teacher model θ can be obtained by function $y = P(x, \theta) = (i, i - 1)$, where $i \in [0, 1]$. In the binary classification tasks, the value of $i = 0.5$ is usually used as the cutoff, which means the image is classified as the first class when $i > 0.5i > 0.5$; otherwise, it is classified as the second class. In our task, i is the confidence of predicting that the image is a CG image. When $i > 0.5$, the image is predicted as a CG image, and a pseudo label is combined with the predicted image. Otherwise, the image is predicted as a natural image.

In our teacher/student training strategy, it is required for the student model to learn the prior knowledge from the teacher model, while the student model requires to surpass the teacher model in detection accuracy. When the teacher model predicts unlabeled data, there must be some prediction errors, which lead to the emergence of noisy labels. Therefore, we introduce the "Local-to-Global" strategy for pseudo-label construction. In the "Local-to-Global" strategy, the image x is randomly divided into n blocks x_j, $j \in [1, n]$, and then the teacher model is used to predict the small-sized image blocks x_j; finally the pseudo label for image x will be decided by majority voting according to the n predicted values of x_j. The "Local-to-Global" strategy has strong flexibility for making classification decision no matter if by one-time prediction of the whole image or multiple-times predictions of the image blocks.

3.3. Malicious Attacks. In order to enhance the generalization ability and robustness of the model, kinds of malicious attacks are applied to D' to obtain a noisy dataset D'_a. Figure 2 shows an original image sample with its processed samples after seven kinds of malicious attacks, which include the following:

(1) Noise attack: to add salt and pepper noise or Gaussian noise with SNR $\in (0.9, 1.0)$.

(2) Translation: to move the image in a random given direction with the distance $D \in (0, 50)$.

(3) Uniform scaling: to enlarge or shrink an image, where the scaling ratio is $r \in (0.75, 1.25)$.

(4) Partial content blocking: a square area with size of 5×5 is randomly selected in the image to change the color to be black.

(5) Color channel change: to convert the original RGB image to a grayscale image.

(6) Affine transformation: a geometric transformation to transform the original vector space into another vector space by performing a linear mapping method on it.

(7) Blurring attack: to blur the image by a Gaussian low-pass filter with kernel size $[3, 3]$ and standard deviation $\sigma = 0$.

In our strategy, after pseudo-label dataset generation by using teacher model, malicious attacks will be randomly selected with a random parameter setting in the predefined range and be applied to partial images which are randomly selected from the mixed dataset D' with a certain probability. Here in our experimental tests, the probability of attacked image is set to be 30%. The number of images in the dataset before and after attacks remains constant, but the image content and data distribution information are expanded to enhance the generalization ability and robustness of model training.

3.4. Learning Rate Decay Strategy. Learning rate is an important hyperparameter in neural network training, which can control both the magnitude for model weights updating and the training speed. If the learning rate is too large, the weights learning will fluctuate greatly, and it is hard to get the optimal solution. If the learning rate is too small, it will result in a long training process. Therefore, learning rate setting or adjustment is crucial to neural network training. In our FTTSE strategy, a learning rate decay strategy is implemented while iterative training. Before iteration, a larger learning rate is used to speed up the initial model fitting. Then the learning rate is gradually reduced to prevent the model from falling into a local optimum in training iterations. This operation for learning rate decay is expressed mathematically as

$$lr_k = \frac{1}{10} lr_{k-1}, \tag{1}$$

where lr_k is the learning rate of k^{th} is the round of training iteration.

4. Experimental Results and Analysis

4.1. Details of Experimental Settings. In our experiment, we collected three image datasets for network training and testing, including Columbia dataset [5], DSTok dataset [37], and SPL2018 dataset [23], which are mainstream experimental datasets for current research work on CG image detection. Table 1 compares the three datasets in aspects of the number of natural images, the number of CG image, and the image size range. Besides, we will introduce the sources of image collecting in these datasets and make a simple analysis of the detection difficulty of each dataset.

The full name of Columbia dataset [5] is called Columbia Photographic Images (PIM) and Photorealistic Computer Graphics (PRCG) Dataset, which is collected by Ng et al. and is the earliest public dataset for CG image detection. The dataset contains 800 PRCG images collected from the Internet, 800 photographed PRCG images recaptured with cameras, 800 PIMs from personal collection, and 800 PIMs from Google image search. Because the image dataset has diverse image sources and the number of images for each image type is relatively small, the image classification of this dataset is relatively difficult.

DSTok dataset [37] is constructed by Tokuda et al. which contains 4850 natural images and 4850 computer-generated images. All images are collected from the Internet, including natural images with indoor and outdoor landscapes

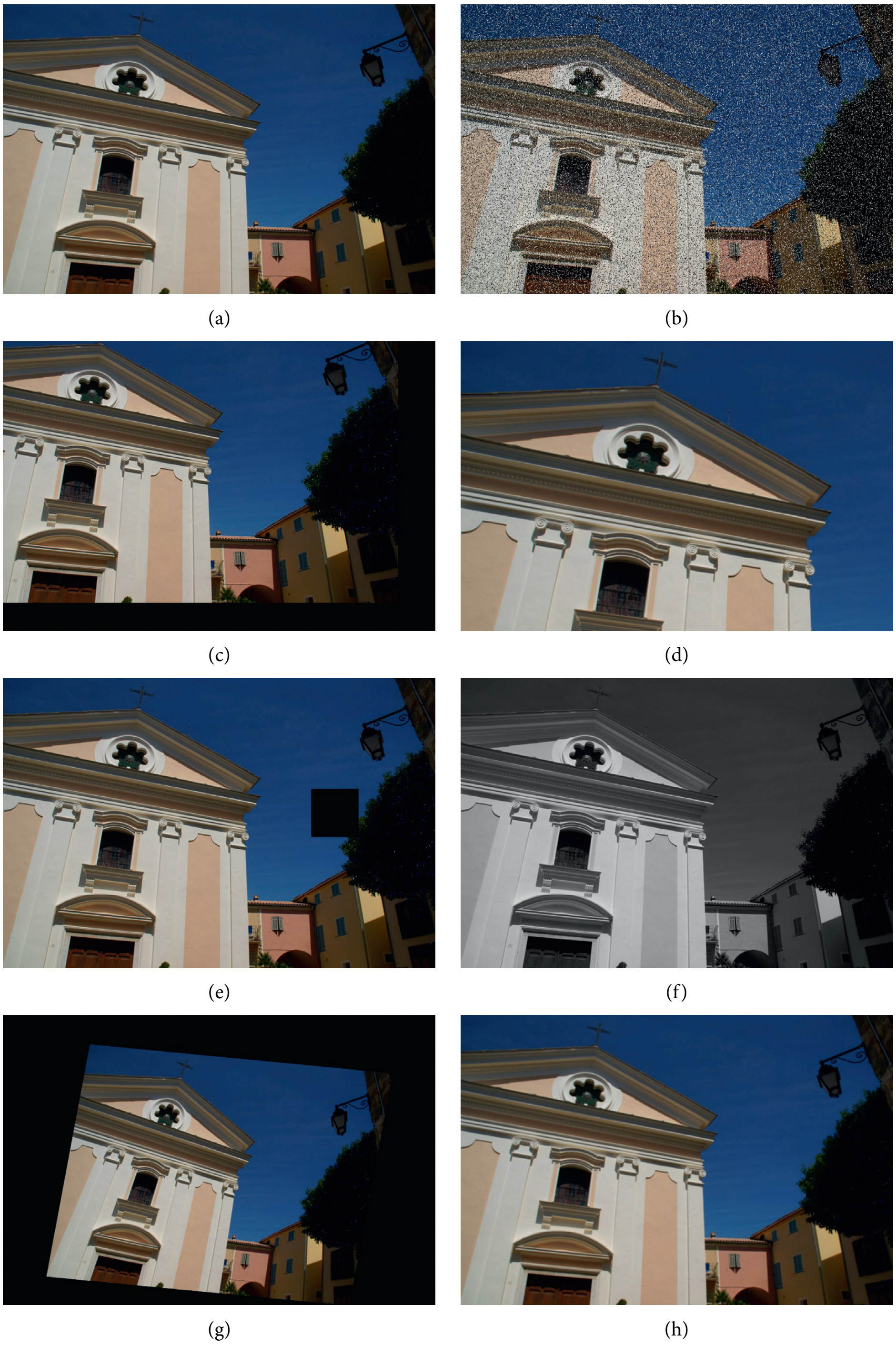

Figure 2: Original image sample and processed samples after different attacks. (a) No attack. (b) Noise. (c) Translation. (d) Scaling. (e) Partial content blocking. (f) Color channel change. (g) Affine transformation. (h) Blurring.

Table 1: Comparison of three datasets implemented in our experiment.

Dataset	Natural image number	CG image number	Image size
Columbia [5]	1600	1600	276 * 421~1398 * 1404
DSTok [37]	4850	4850	609 * 603~3507 * 2737
SPL2018 [23]	6800	6800	266 * 199~2048 * 3200

captured by various devices, and CG images collected with more content subjects, such as characters, architectures, and landscapes. DSTok dataset has a large number of images with comprehensive content categories, and it is an important dataset for CG image detection research.

SPL2018 dataset [23] is constructed by He et al. with 6800 CG images and 6800 natural images. Besides the images collected from the Internet, there are some CG images generated by more than 50 rendering software, e.g., *3DS Max*

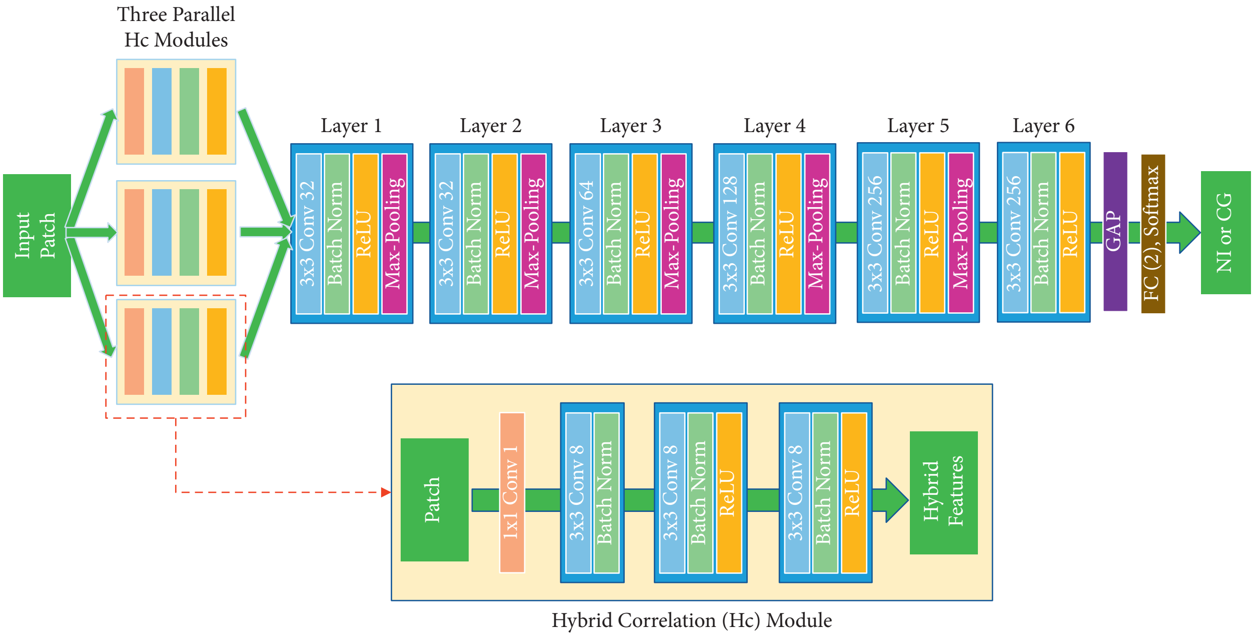

Figure 3: ScNet network structure proposed in [27].

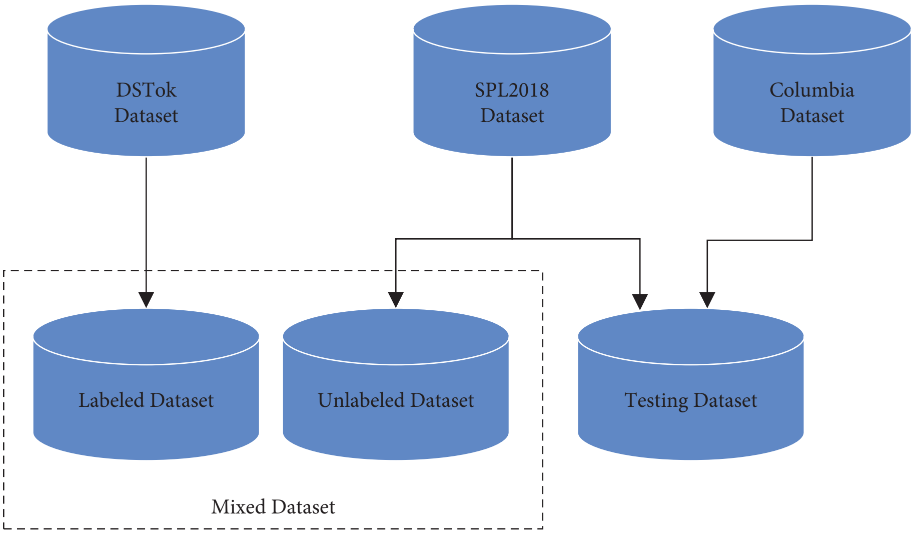

Figure 4: Division of different datasets.

and *Maya*. Natural images are photos captured by different types of cameras in various scenes. Besides the diverse image content subjects, the SPL2018 dataset contains images with different resolutions, especially the low-resolution images, which is more suitable to test the classification performance with more experimental scenarios requirements.

In our experiments, we use a convolutional-neural-network based model named *ScNet* proposed by Zhang et al. [27]. The key part of *ScNet* is a network structure called the self-coding module, which is efficient for deep learning of the correlation among three color channels and pixel-related features. Figure 3 shows the network structure diagram of *ScNet*. In our experiments, we compare the stability, generality, and robustness of the *ScNet* model before and after using our proposed FTTSE training strategy.

There are three kinds of dataset constructed for teacher and student model training in our work, including labeled dataset, unlabeled dataset, and testing dataset. For labeled data, we selected all images from the DSTok dataset, which is one of the most used with critical image quality in the field of CG image detection. For unlabeled dataset, we randomly selected 5000 CG images and 5000 natural images from the SPL2018 dataset and removed their labels. Then the remaining 3600 images in SPL2018 dataset and 3200 images in Columbia dataset are selected for testing dataset to verify the detection performance of the model. Figure 4 shows the dataset division process in our experiment.

For the preprocessing of image samples in the dataset, we randomly crop 20 image batches with size 224 × 224 for each sample and then randomly divide these cropped image batches into training set, validation set, and testing set with a ratio of 8: 8: 1. In iterative training, we use a batch size of 32, including 16 CG images and 16 natural images. During training, the CNN parameters are optimized by stochastic gradient descent. The initial learning rate is set as 0.001, and the order of the training set is randomly shuffled after each epoch. There are four models obtained by network training at different stages of our self-training process. The first teacher model (*M*0) is obtained by the first initialization training in the labeled dataset. The training process of *M*0 stops after 120 epochs. The first student model (*M*1) is obtained by student initialization training in the mixed dataset, which contains the labeled and the pseudo-labeled image samples. The pseudo labels are predicted by *M*0. After generating *M*0 and *M*1, there are two rounds of iteration for retraining student model: *M*2 and *M*3 with attacked mixed dataset. The training processes of *M*2 and *M*3 carry out the FTTSE strategy with learning rate decay and stop after 60 epochs.

For these four models mentioned above, we designed four experiments to evaluate the model stability, benchmark the four models, validate the model generality, and evaluate the model robustness, respectively. In order to fully reflect the classification ability of the testing model, there are three experimental indicators calculated for model evaluation in terms of detection accuracy, precision, and recall. Due to the randomness of the detection results of *ScNet* model, our FTTSE self-training experiment was repeated for three times, and the average results were finally calculated to verify the network performance. All experiments are implemented on a GeForce GTX 1080Ti using the deep learning framework PyTorch0.4.1.

4.2. Stability of Training Model. In the self-training process, the model *M*0 is actually similar as the model proposed by Zhang et al. [27] only with small changes in our desired experimental condition settings. Compared with the model of [27], we use the same *ScNet* network but under a different image dataset with larger size of image scene and smaller epochs of training. Following the training settings mentioned in Section 4.1, the self-training process is repeated three times in DSToK dataset to validate the network stability. For each time, the four models are evaluated by calculating their detection rates of accuracy, precision, and recall. Tables 2–4 show the experimental results of the three-times repeated self-training process, respectively. The average detection results are calculated and shown in Table 5.

As shown in Tables 2–5, the experimental results of *M*0 are basically consistent with the simulation results in [27]. In our experiment, the average detection accuracy of *M*0 approximately reaches 94.79%. This proves the validation of *ScNet* for CG image detection. In addition, the experimental results of *M*1, *M*2, and *M*3 in the three times experiments keep a stable performance even if the training image samples are maliciously attacked. The four trained models shown in Table 3 has the best performance results among the three-times repeated experiments. Compared with *M*0, the detection accuracy of *M*2 is improved by 0.58%, whereas the detection accuracy of *M*3 compared with *M*1 is improved by 0.37%. In the whole self-training process, the final training model *M*3 is improved compared with the initial training model *M*0 with an accuracy rate that increased by 0.88%, precision rate increased by 0.97%, and recall rate increased by 0.80%. The experimental results in Tables 2–5 reveal the stability of the model trained by our proposed FTTSE strategy and the performance improvement in CG image detection.

Table 2: Results of stability test experiment I.

Model	Accuracy (%)	Precision (%)	Recall (%)
*M*0	94.85	93.91	95.83
*M*1	94.51	93.14	95.90
*M*2	94.45	94.34	95.26
*M*3	94.54	94.15	95.01

Table 3: Results of stability test experiment II.

Model	Accuracy (%)	Precision (%)	Recall (%)
*M*0	94.36	92.79	95.96
*M*1	94.87	93.20	95.56
*M*2	94.94	93.42	96.48
*M*3	95.24	93.76	96.76

Table 4: Results of stability test experiment III.

Model	Accuracy (%)	Precision (%)	Recall (%)
*M*0	95.16	94.39	96.37
*M*1	95.29	93.57	97.41
*M*2	95.07	93.43	97.13
*M*3	95.48	93.94	97.40

Table 5: Average results of three experiments.

Model	Accuracy (%)	Precision (%)	Recall (%)
*M*0	94.79	93.70	96.05
*M*1	94.89	93.57	96.29
*M*2	94.82	93.30	96.29
*M*3	95.09	93.95	96.39

4.3. Benchmarking Test. Here we use the four models trained by the second experiment with the best performance as shown in Section 4.2 for benchmarking test. The remaining 3600 image samples in SPL2018 dataset with their labels, which are not used for training models, will be used as the testing set to benchmark the four models. The test results of initial training teacher model *M*0 is used as the baseline for benchmarking, since there is no image sample in SPL2018 used for *M*0 training. The detection accuracy, precision, and recall of the four models on remaining SPL2018 image samples are compared in Table 6. As can be seen in Table 6, the final training model *M*3 performs stably higher even if there are malicious attacks applied to the training images. Due to the prior knowledge of *M*0, *M*1, and *M*2, the final model *M*3 has ability of quickly learning the diagnostic features for CG image detection. Compared with the initial training teacher model *M*0, the detection accuracy of *M*3 is improved by 5.18%. The detection ability of the four models shows an upward trend in all terms of accuracy, precision, and recall. The good verification results shown in Table 6 illustrate the model improvement using FTTSE strategy.

4.4. Generality Evaluation Test. For validating the model generality, the image samples in Columbia dataset are used as unknown samples for testing detection accuracy, which were not used in any model training process. In this experiment, both 1600 natural images and 1600 CG images in the Columbia dataset are used to validate the generality of the four models to distinguish CG images from natural images.

The detection accuracy, precision, and recall of the four models on Columbia dataset are compared in Table 7. All the experimental results are visualized by bar-chart as shown in Figure 5. According to the experimental results, with the iteration of FTTSE training, the retrained model $M1$ compared with its prior fine-tuned teacher model $M0$ is improved in all terms of detection accuracy, precision, and recall. Likewise, the retrained model $M3$ is improved compared with its prior fine-tuned teacher model $M1$. The experimental results shown in Table 7 and Figure 5 validate the generalization ability of the four models, while it is also proved that the FTTSE self-training strategy can effectively strengthen the generality of model by iterative training.

4.5. Robustness Test. Since the malicious attacks are added in our iterative training process as introduced in Section 3.3, the model after implementing our teacher/student exchange strategy already has a certain robustness against various attacks. In order to sufficiently evaluate the robustness of the network model, here we enhance the attack strength or expand the subcategories of attack in our experiment. The seven kinds of malicious attack applied for robustness evaluation are reset as follows.

(1) Noise attack with SNR $\in (0.8, 1.0)$

(2) Translation with moving distance $D \in (0, 100)$

(3) Uniform scaling with scaling ratio $r \in (0.5, 1.5)$

(4) Partial content blocking with the blocking area size of 10×10

(5) Color channel change with adding HSV color space processing

(6) Affine transformation same as before

(7) Blurring attack with adding media filtering with kernel size $[3, 3]$

Here we still use the four models trained from the second experiment to evaluate their robustness against strengthened attacks on the remaining 3600 image samples in SPL2018 dataset. For the testing images, we design two experiments where the strengthened attacks are randomly selected and applied to 50% and 100% of images, respectively. The experimental results are shown in Tables 8 and 9, which are visualized in Figures 6 and 7.

Table 6: Benchmark test results.

Model	Accuracy (%)	Precision (%)	Recall (%)
$M0$	83.42	87.76	80.74
$M1$	87.83	90.91	85.62
$M2$	88.54	91.30	86.51
$M3$	88.60	91.62	86.39

Table 7: Generality evaluation test results.

Model	Accuracy (%)	Precision (%)	Recall (%)
$M0$	71.09	76.76	68.95
$M1$	73.15	83.31	69.24
$M2$	72.55	80.15	69.58
$M3$	74.17	84.98	69.87

As shown in Tables 8 and 9, the detection accuracy of the initial training teacher model $M0$ only achieves 71.93% and 73.72% with 50% and 100% attacks, respectively. The detection performance of $M0$ rapidly decreased compared with the benchmark experimental results in Section 4.3 without image attacks. By introducing malicious attacks in our training strategy, the models $M1$, $M2$, and $M3$ generally perform an upward trend in the three indicators of detection accuracy, precision, and recall as shown in Figures 6 and 7. In Table 9, the final model $M3$, compared with $M0$, performs a higher detection rate with an increase of 7.63%, 6.18%, and 7.04% in terms of detection accuracy, precision, and recall. By analyzing the experimental results above, the improvement indicates our proposed FTTSE strategy can effectively enhance the robustness of the model.

4.6. Analysis of Different Dataset-Combination Settings. In the previous experimental initialization settings, our dataset-combination setting is to select all images from the DSTok dataset as labeled data, randomly select 5000 CG images and 5000 natural images from the SPL2018 dataset as unlabeled data, and select the remaining 3600 images in SPL2018 dataset and 3200 images in Columbia dataset as testing dataset. Under this dataset-combination setting, the proposed method presents good performance in our experiments as shown in Sections 4.2–4.5. In order to verify the stability of our improved FTTSE strategy with different dataset-combination settings, we will further analyze the CG image detection accuracy of the models trained by different dataset-combination settings in this section. Here the previous dataset-combination setting is marked as “Comb1.” Besides, we add two different dataset-combination settings marked as “Comb2” and “Comb3,” respectively. In the experiment of Comb2, all the images in Columbia dataset are selected as the labeled data, 5000 CG images and 5000 natural images are randomly selected from the SPL2018 dataset as the unlabeled data, and the remaining 3600 images in SPL2018 dataset and all the images in DSTok dataset are selected as the testing dataset. In the experiment of Comb3, 5000 CG images and 5000 natural images in SPL2018 dataset are selected as the labeled data, all the images in DSTok dataset are selected as the unlabeled data, and the remaining

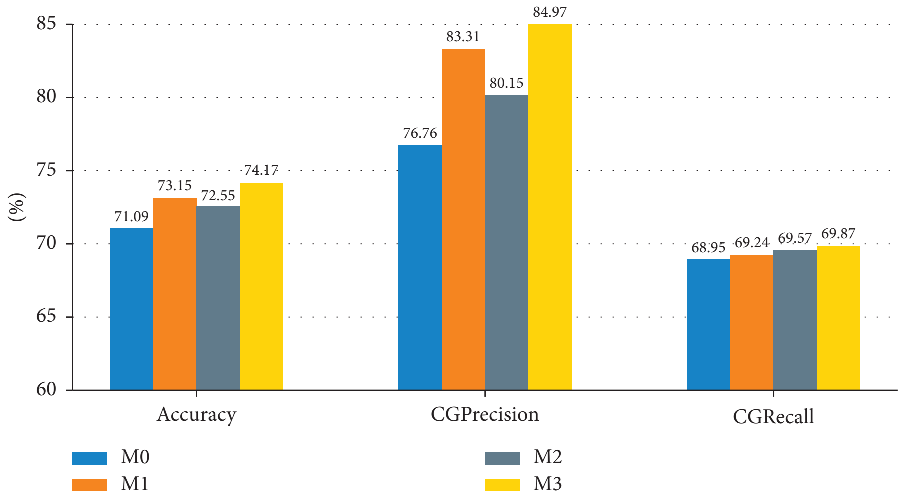

Figure 5: Generality evaluation test results.

Table 8: Robustness test results (with attack probability 50%).

Model	Accuracy (%)	Precision (%)	Recall (%)
*M*0	71.93	86.40	66.99
*M*1	84.40	91.00	80.38
*M*2	84.85	90.09	81.53
*M*3	85.12	90.84	81.51

Table 9: Robustness test results (with attack probability 100%).

Model	Accuracy (%)	Precision (%)	Recall (%)
*M*0	73.72	83.39	69.87
*M*1	80.98	91.02	75.78
*M*2	81.29	89.02	77.09
*M*3	81.35	89.57	76.91

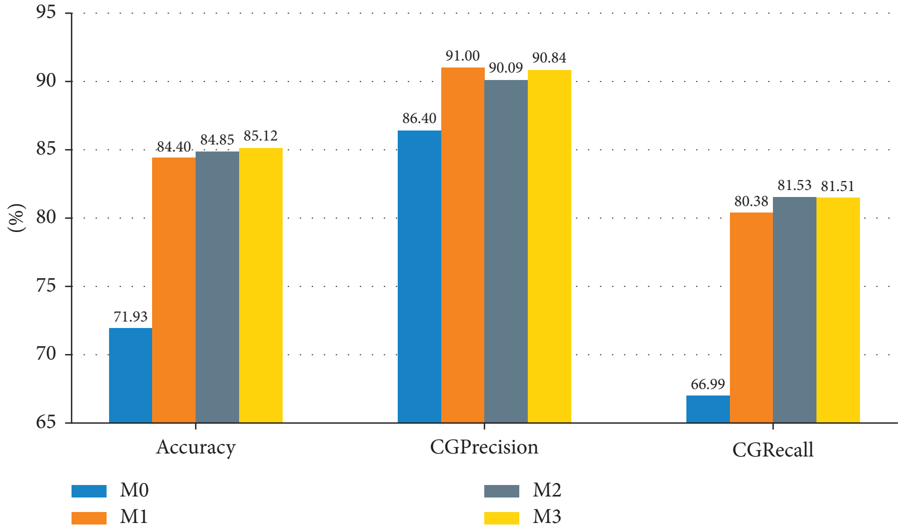

Figure 6: Robustness test results (with attack probability 50%).

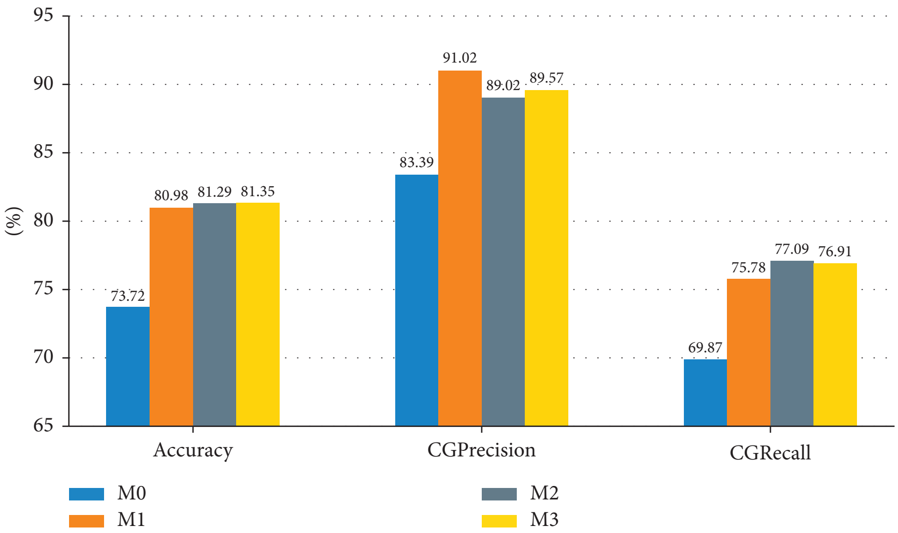

Figure 7: Robustness test results (with attack probability 100%).

Table 10: Test results for different dataset-combination settings without any attacks to the testing dataset.

Model	Accuracy (%)			Precision (%)			Recall (%)		
	Comb1	Comb2	Comb3	Comb1	Comb2	Comb3	Comb1	Comb2	Comb3
*M*0	83.42	81.36	95.40	87.76	79.04	94.75	80.74	82.86	96.00
*M*1	87.83	85.92	95.18	90.91	84.37	94.64	85.62	87.11	95.67
*M*2	88.54	87.85	95.16	91.30	87.33	94.91	86.51	88.24	95.39
*M*3	88.60	88.27	95.10	91.62	87.64	94.47	86.39	88.74	95.65

Table 11: Test results for different dataset-combination settings with attacks to 50% of the testing dataset.

Model	Accuracy (%)			Precision (%)			Recall (%)		
	Comb1	Comb2	Comb3	Comb1	Comb2	Comb3	Comb1	Comb2	Comb3
*M*0	71.93	75.44	82.13	86.40	77.44	96.54	66.99	74.45	74.93
*M*1	84.40	82.65	91.73	91.00	83.56	94.83	80.38	82.05	89.28
*M*2	84.85	85.04	91.25	90.09	86.44	94.96	81.53	84.08	88.39
*M*3	85.12	84.22	91.64	90.84	83.85	94.60	81.51	84.46	89.30

Table 12: Test results for different dataset-combination settings with attacks to 100% of the testing dataset.

Model	Accuracy (%)			Precision (%)			Recall (%)		
	Comb1	Comb2	Comb3	Comb1	Comb2	Comb3	Comb1	Comb2	Comb3
*M*0	73.72	70.62	70.97	83.39	76.72	97.57	69.87	67.99	63.68
*M*1	80.98	79.82	88.06	91.02	83.43	94.82	75.78	77.80	83.52
*M*2	81.29	82.34	87.19	89.02	86.03	94.92	77.09	80.10	82.20
*M*3	81.35	80.75	88.16	89.57	81.40	94.75	76.91	80.34	83.71

3600 images in SPL2018 dataset and all the images in Columbia dataset are selected as the testing dataset.

In each dataset-combination setting, four models are trained by the FTTSE strategy, and the detection ability of each model is benchmarked on the images in the testing dataset without any attack, with 50% malicious attack and 100% malicious attack, respectively. The attack setting is the same as introduced in Section 4.5, and the detection ability is calculated in terms of accuracy, precision, and recall, respectively. All the experimental results for different dataset-combination settings are shown in Tables 10–12.

As shown in Table 10, the detection accuracy rates for Comb1 and Comb2 settings without any attacks are both improved from 83.42% and 81.36% to 88.60% and 88.27%. As the labeled training images and the testing images are both selected from SPL2018 in Comb3 setting, the detection accuracy performs significantly superior than that of Comb1 and Comb2 settings, and all the detection accuracy rates of the four models keep higher than 95%. For the robustness test shown in Tables 11 and 12, it can be seen that the detection accuracy of the initial training teacher model $M0$ is decreased significantly with 50% and 100% attacks, which is one of the difficult problems faced by deep learning. After the teacher/student iterative training by our FTTSE strategy, the detection accuracy of $M3$ achieves 8%~13% higher than $M0$ with 50% attack for testing images and 7%~17% higher than $M0$ with 100% attack.

By briefly glancing at Tables 10–12, it can be seen that (1) our proposed FTTSE strategy can maintain good detection performance facing with different dataset-combination settings and (2) the proposed FTTSE strategy can enhance the robustness of the model with a significant effect in detection accuracy. In addition, it is noteworthy that in Comb2 dataset-combination setting, the number of labeled images in Columbia dataset is relatively small, but the detection capability and robustness performance with the same number of unlabeled image and the same pseudo-label construction strategy for model training can still keep stable improvement in the experiment, which further proves that our method is an effective solution to the problem of lack of labeled training samples in deep learning.

5. Conclusion

This paper proposes an improved self-training model with FTTSE strategy to distinguish CG images from naturally captured images. We improve the CG image detection accuracy of existing model through designing the new teacher/student exchange strategy, pseudo-label construction strategy, malicious-attack strategy, and learning rate decay strategy. Our experimental results show that (1) the stability of the trained model using our proposed FTTSE strategy keeps a good performance for image classification; (2) the detection accuracy of the proposed model is improved by 5.18% after iterative training; (3) the robustness of the model is improved; i.e., even if the testing image set is faced with various malicious attacks, the self-training model can still show good detection accuracy which is improved by 7.63% after iterative training.

However, during the iterative training in the experiments, the mixed training dataset will be constructed with wrong pseudo labels due to the prediction errors. The errors will be propagated and amplified with the iterative training. This causes the label distribution in mixed dataset extremely uneven in the subsequent rounds of iterative training, and the CG image detection accuracy is declining continuously. In the future work, the methods about how to overcome the imbalanced training samples in the self-training strategy and how to further improve the image classification ability will be further studied.

Data Availability

Some data and the key partial code used during the study appearing in the submitted article are available by email contacting the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] O. Alexander, M. Rogers, W. Lambeth et al., "The digital emily project: achieving a photorealistic digital actor," *IEEE Computer Graphics and Applications*, vol. 30, no. 4, pp. 20–31, 2010.

[2] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Niessner, "Face2face: real-time face capture and reenactment of rgb videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2387–2395, IEEE, Las Vegas, NV, USA, June 2016.

[3] S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Synthesizing Obama," *ACM Transactions on Graphics*, vol. 36, no. 4, pp. 1–13, 2017.

[4] T.-T. Ng, S.-F. Chang, J. Hsu, L. Xie, and M.-P. Tsui, "Physics-motivated features for distinguishing photographic images and computer graphics," in *Proceedings of the 13th Annual ACM International Conference on Multimedia*, pp. 239–248, Singapore, November 2005.

[5] T.-T. Ng, S.-F. Chang, J. Hsu, and P. Martin, "Columbia photographic images and photorealistic computer graphics dataset," pp. 205–2004, Columbia University, New York, NY, USA, 2005, ADVENT Technical Report.

[6] W. Chen, Y. Q. Shi, and G. Xuan, "Identifying computer graphics using hsv color model and statistical moments of characteristic functions," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, pp. 1123–1126, IEEE, Beijing, China, July 2007.

[7] A. C. Gallagher and T. Chen, "Image authentication by detecting traces of demosaicing," in *Proceedings of the 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1–8, IEEE, Anchorage, Alaska, USA, June 2008.

[8] R. Zhang, R.-D. Wang, and T.-T. Ng, "Distinguishing photographic images and photorealistic computer graphics using visual vocabulary on local image edges," in *Proceedings of the International Workshop on Digital-Forensics and Watermarking*, pp. 292–305, Springer, Atlantic, NJ, USA, October 2011.

[9] J. Wang, T. Li, Y.-Q. Shi, S. Lian, and J. Ye, "Forensics feature analysis in quaternion wavelet domain for distinguishing photographic images and computer graphics," *Multimedia Tools and Applications*, vol. 76, no. 22, pp. 23721–23737, 2017.

[10] E. R. S. de Rezende, G. C. S. Ruppert, A. Theóphilo, E. K. Tokuda, and T. Carvalho, "Exposing computer generated images by using deep convolutional neural networks," *Signal Processing: Image Communication*, vol. 66, pp. 113–126, 2018.

[11] N. Rahmouni, N. Vincent, J. Yamagishi, and I. Echizen, "Distinguishing computer graphics from natural images using convolution neural networks," in *Proceedings of the IEEE International Workshop on Information Forensics and Security (WIFS)*, pp. 1–6, IEEE, Rennes, France, December 2017.

[12] W. Quan, K. Wang, D.-M. Yan, and X. Zhang, "Distinguishing between natural and computer-generated images using convolutional neural networks," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2772–2787, 2018.

[13] Q. Xie, M.-T. Luong, E. Hovy, and V. L. Quoc, "Self-training with noisy student improves imagenet classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision*

and Pattern Recognition, pp. 10687–10698, Seattle, WA, USA, June 2020.

[14] K. B. Meena and V. Tyagi, “A deep learning based method to discriminate between photorealistic computer generated images and photographic images,” in *Proceedings of the International Conference on Advances in Computing and Data Sciences (ICACDS2020): Advances in Computing and Data Sciences*, pp. 212–223, Springer, Valletta, Malta, April 2020.

[15] K. Rajasekhar and G. Indra Sai Kumar, “Recognition of natural and computer-generated images using convolutional neural network,” in *Proceedings of the Advances in Communications, Signal Processing, and VLSI*, pp. 11–21, Springer, Hyderabad, India, April 2021.

[16] Y. Zhu, F. Zhuang, J. Wang, J. Chen, J. Bian, and H. Xiong, “Deep subdomain adaptation network for image classification,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 4, pp. 1713–1722, 2020.

[17] Z. Li, J. Ye, and Y. Q. Shi, “Distinguishing computer graphics from photographic images using local binary patterns,” in *Proceedings of the 2021 International Workshop on Digital-Forensics and Watermarking (IWDW2012)*, pp. 228–241, Springer, Shanghai, China, October 2013.

[18] R. Wu, X. Li, and B. Yang, “Identifying computer generated graphics via histogram features,” in *Proceedings of the 18th IEEE International Conference on Image Processing (ICIP2011)*, pp. 1933–1936, IEEE, Brussels, Belgium, September 2011.

[19] S. Lyu and H. Farid, “How realistic is photorealistic?” *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 845–850, 2005.

[20] W. Bai, Z. Zhang, B. Li et al., “Robust texture-aware computer-generated image forensic: benchmark and algorithm,” *IEEE Transactions on Image Processing*, vol. 30, pp. 8439–8453, 2021.

[21] G. Gando, T. Yamada, H. Sato, S. Oyama, and M. Kurihara, “Fine-tuning deep convolutional neural networks for distinguishing illustrations from photographs,” *Expert Systems with Applications*, vol. 66, pp. 295–301, 2016.

[22] Y. Yao, W. Hu, W. Zhang, T. Wu, and Y.-Q. Shi, “Distinguishing computer-generated graphics from natural images based on sensor pattern noise and deep learning,” *Sensors*, vol. 18, no. 4, p. 1296, 2018.

[23] P. He, X. Jiang, T. Sun, and H. Li, “Computer graphics identification combining convolutional and recurrent neural networks,” *IEEE Signal Processing Letters*, vol. 25, no. 9, pp. 1369–1373, 2018.

[24] S. Sabour, N. Frosst, and G. E. Hinton, “Dynamic routing between capsules,” in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 3856–3866, Long Beach, CA, USA, December 2017.

[25] H. H. Nguyen, J. Yamagishi, and I. Echizen, “Capsule-forensics: using capsule networks to detect forged images and videos,” in *Proceedings of the 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP2019)*, pp. 2307–2311, IEEE, Brighton, UK, May 2019.

[26] D. B. Tarianga, P. Senguptab, A. Roy, R. Subhra Chakraborty, and R. Naskar, “Classification of computer generated and natural images based on efficient deep convolutional recurrent attention model,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPR)*, pp. 146–152, Las Vegas, NV, USA, July 2019.

[27] R.-S. Zhang, W.-Z. Quan, L.-B. Fan, L.-M. Hu, and D.-M. Yan, “Distinguishing computer-generated images from natural images using channel and pixel correlation,” *Journal of Computer Science and Technology*, vol. 35, no. 3, pp. 592–602, 2020.

[28] W. Quan, K. Wang, D.-M. Yan, X. Zhang, and D. Pellerin, “Learn with diversity and from harder samples: improving the generalization of cnn-based detection of computer-generated images,” *Forensic Science International: Digital Investigation*, vol. 35, Article ID 301023, 2020.

[29] R. Huang, F. Fang, H. H. Nguyen, J. Yamagishi, and I. Echizen, “A method for identifying origin of digital images using a convolution neural network,” in *Proceedings of the 2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 1293–1299, IEEE, Tokyo, Japan, December 2020.

[30] K. B. Meena and V. Tyagi, “Distinguishing computer-generated images from photographic images using two-stream convolutional neural network,” *Applied Soft Computing*, vol. 100, Article ID 107025, 2021.

[31] B. Chen, W. Tan, G. Coatrieux, Y. Zheng, and Y.-Q. Shi, “A serial image copy-move forgery localization scheme with source/target distinguishment,” *IEEE Transactions on Multimedia*, vol. 23, pp. 3506–3517, 2020.

[32] B. Chen, X. Liu, Y. Zheng, G. Zhao, and Y.-Q. Shi, “A robust gan-generated face detection method based on dual-color spaces and an improved xception,” *IEEE Transactions on Circuits and Systems for Video Technology*, 2021, https://ieeexplore.ieee.org/document/9552855.

[33] C. Olivier, B. Scholkopf, and Z. Alexander, “Semi-supervised learning,” *IEEE TransActions on Neural Networks*, vol. 20, no. 3, p. 542, 2006.

[34] S. Mukherjee and A. H. Awadallah, “Uncertainty-aware self-training for text classification with few labels,” 2020, https://arxiv.org/abs/2006.15315.

[35] Y. Zou, Z. Yu, X. Liu, B. V. K. Kumar, and J. Wang, “Confidence regularized self-training,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5982–5991, Seoul, Korea, October 2019.

[36] W. Chen, K. Sohn, C. Mellina, A. Yuille, and Y. Fan, “Crest: a class-rebalancing self-training framework for imbalanced semi-supervised learning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10857–10866, Nashville, TN, USA, June 2021.

[37] E. Tokuda, H. Pedrini, and A. Rocha, “Computer generated images vs. digital photographs: a synergetic feature and classifier combination approach,” *Journal of Visual Communication and Image Representation*, vol. 24, no. 8, pp. 1276–1292, 2013.