electronics

Article

Image Authentication and Restoration Using Block-Wise Variational Automatic Encoding and Generative Adversarial Networks

Chin-Feng Lee [1,*], Chin-Ting Yeh [2,†], Jau-Ji Shen [2,†] and Taeshik Shon [3,†]

[1] Department of Information Management, Chaoyang University of Technology, Taichung 41349, Taiwan
[2] Department of Management Information Systems, National Chung Hsing University, Taichung 40277, Taiwan; mj0684521300@gmail.com (C.-T.Y.); jjshen@nchu.edu.tw (J.-J.S.)
[3] Department of Cybersecurity, Ajou University, Suwon 16499, Republic of Korea; tsshon@ajou.ac.kr
* Correspondence: lcf@cyut.edu.tw; Tel.: +886-423323000 (ext. 4293)
† These authors contributed equally to this work.

Abstract: The Internet is a conduit for vast quantities of digital data, with the transmission of images being especially prevalent due to the widespread use of social media. However, this popularity has led to an increase in security concerns such as image tampering and forgery. As a result, image authentication has become a critical technology that cannot be overlooked. Recently, numerous researchers have focused on developing image authentication techniques using deep learning to combat various image tampering attacks. Nevertheless, image authentication techniques based on deep learning typically classify only specific types of tampering attacks and are unable to accurately detect tampered images or indicate the precise location of tampered areas. The paper introduces a novel image authentication framework that utilizes block-wise encoding through Variational Autoencoder and Generative Adversarial Network models. Additionally, the framework includes a classification mechanism to develop separate authentication models for different images. In the training phase, the image is first divided into blocks of the same size as training data. The goal is to enable the model to judge the authenticity of the image by blocks and to generate blocks similar to the original image blocks. In the verification phase, the input image can detect the authenticity of the image through the trained model, locate the exact position of the image tampering, and reconstruct the image to ensure the ownership.

Keywords: deep learning; Generative Adversarial Network; Variational Autoencoder; image authentication; image attack; image recovery; tampering and positioning

Citation: Lee, C.-F.; Yeh, C.-T.; Shen, J.-J.; Shon, T. Image Authentication and Restoration Using Block-Wise Variational Automatic Encoding and Generative Adversarial Networks. *Electronics* **2023**, *12*, 3402. https://doi.org/10.3390/electronics12163402

Academic Editor: Stefanos Kollias

Received: 3 July 2023
Revised: 1 August 2023
Accepted: 2 August 2023
Published: 10 August 2023

1. Introduction

With the popularity of social networks nowadays, all ages from elementary school students to elders will use social media to convey messages to connect with feelings. With the advent of the 5G era, with the characteristics of high bandwidth, low latency, and wide connectivity, the circulation of digital data such as text, pictures, voice, and video will usher in unprecedented frequency and intensity. Especially under the rapid development of new media, the transmission of images is better than text and voice, because it can quickly attract attention and can more effectively convey the meaning you want to express. However, the increasingly serious problem of digital copyright is also increasing with the evolution of the background of time and space. Therefore, ensuring the ownership, integrity, and correctness of images is a technology that is urgently needed today, and it is also the direction of many examples of research to prevent tampering and misappropriation by interested parties.

Image authentication can be divided into two categories: active authentication and passive authentication [1]. Active authentication is to embed a known verification code

in the image content before sending the image through a public channel that may be attacked, and to compare the previously embedded verification code as the basis for authentication. However, this method requires special hardware equipment and software tools to embed the verification code in the image before sending the image to the receiving end. The passive authentication method is that the receiving end directly uses the received image to evaluate the authenticity or integrity of its content and does not need to use the signature or watermark of the original image of the sending end. The core idea of passive authentication is based on the fact that digital forgery may not leave any visual clues that have been tampered with, but it may interfere with potential statistical attributes or the image consistency of natural scenery images, thereby processing new forged images. The assumptions that lead to inconsistencies in various forms of images indicate that these inconsistencies can be used for forgery detection. Therefore, the popularity of passive authentication is that it does not require any prior information about the image and can distinguish various traces of tampering.

Common methods in active authentication are cryptography-based verification technology (Cryptography) and digital watermark technology (Digital Watermark). Password-based verification technology [2–5] uses a hash function to calculate the identity verification password and compares the identity verification password to know whether the image has been tampered with. The disadvantage of password-based authentication technology is that we cannot identify the area of image tampering. Watermarking technology can be roughly divided into robust watermarking, semi-fragile watermarking, and fragile watermarking [6–11]. The robust watermark technology can still take out the watermark hidden in the watermark image after it is subjected to common noise processing or image compression processing. It has the characteristics of not being easily destroyed, so it can be used to prove copyright and intellectual property rights. The fragile watermark is the watermark hidden in the image, which is easily destroyed by tampering, and therefore the tampered area can be accurately detected. The semi-fragile watermark is somewhere in between and is highly sensitive to tampering, so it is often used in image tampering detection applications.

Today, when cameras, mobile phones, monitors, driving recorders, and other equipment are ubiquitous, the quantity of video data in human society has become unprecedentedly huge. Computer vision has been developed for a long time. Recently, it has taken off due to deep learning, which has derived various important image recognition applications, such as face recognition, object detection, vehicle tracking, street view analysis, etc. In 1989, Yann LeCun published a convolutional neural network for digital recognition of handwritten postal codes, and in 1998 proposed the first well-known convolutional neural network architectureLeNet-5 [12]. LeNet-5 is used to identify handwritten digital data, which reflects the ability of convolutional neural network (CNN) to understand image content and extract effective features. In the future, more CNN models with different architectures and higher accuracy will be derived, laying an important foundation for today's image recognition. Recently, image authentication technologies based on convolutional neural networks have been derived [13–16]. Compared with the traditional detection method, the analysis is only based on the feature changes caused by a certain specific image operation. The advantage of the convolutional neural network is that it has powerful feature extraction capabilities, can learn more advanced image semantic information, and accurately reflect the essential characteristics of the data, so it is more conducive to the analysis of the results. Image authentication using CNN is a passive authentication method.

In real image recognition applications, for example, neural networks usually need to analyze tens of thousands of photos before learning how to recognize them. However, this requires humans to carefully mark the content on each photo before these tagged photos can be used to train neural networks, which is a laborious and time-consuming task. In 2014, Ian Goodfellow proposed the Generative Adversarial Network (GAN) [17]. Unlike in the past, training neural networks often requires a supervised machine training model that uses humans to provide a large amount of labeled data. The Generative Adversarial

Network (GAN) generates a large amount of training data through a small amount of real data, which is like an "unsupervised" model. A system that learns more knowledge with less human assistance is created. GAN can not only greatly reduce the demand for training data, but also provide a better method for unsupervised machine learning through its own generation network and discrimination network that compete with each other. It is a major development of Neural Network. GANs are often recognized for their ability to generate highly realistic images but suffer from two major limitations in comparison to VAEs: mode collapse and the absence of an encoder network. To leverage both the advantages of GANs and VAEs, the combination of a Generative Adversarial Network (GAN) and a Variational Autoencoder (VAE) has been explored in recent state-of-the-art research [18]. Accordingly, many variations and extensions of VAE-GANs have been proposed in kinds of applications. For example, Mina Razghandi et al. proposed a scheme based on the Variational AutoEncoder Generative Adversarial Network (VAE-GAN) [19] to generate synthetic time series data for smart homes. The article indicate that the distribution of synthetic data generated by VAE-GAN exhibits the highest level of comparability to that of real data.

At present, GAN is mostly used in generating data, such as image and audio-visual generation, synthesis, identification, restoration, etc. However, relatively few people in the relevant literature currently use GAN for image authentication, and most of them use GAN to generate specific images according to specific needs and optimize the generalization ability of their detection models. We believe that the characteristics of the generation network and the discrimination network against learning will certainly help as a prototype of a detection tool to spot forged images, but only if it is to be able to detect the authenticity of the image during the image authentication process. In addition to the comprehensive capabilities of tampering positioning and image restoration, some improvements must be made to the original GAN framework. Therefore, this research focuses on using the powerful ability of deep learning to extract features to design a new image authentication framework based on image blocks. We propose a novel block-level image authentication mechanism that combines VAE's coding model and a GAN model as the base architecture, and adds a classification mechanism to enhance the effectiveness of image authentication. The proposed image authentication method is called as VAE-bGANs for short. Through this VAE-bGANs model, we mainly want to detect the authenticity of the image, locate the exact location of the image tampering, and recover the tampered area to reconstruct the original image.

This research makes the following contributions:

1. We utilized the VAE coding model and GAN model as the base architecture and incorporated a classification mechanism, resulting in a novel image authentication architecture. This architecture is capable of handling multiple tampering attacks even with limited training data.
2. By integrating a classification mechanism, our model achieves enhanced performance. Our method enables the restoration of tampered images and reconstruction of the original images.
3. Our approach employs block-based analysis to detect image authenticity. By combining the detection results of individual blocks with residual map calculations, we are able to accurately locate tampering areas.

The remainder of this paper is structured as follows. Section 2 briefly reviews image authentication methods related to deep learning techniques. Section 3 introduces the proposed method in detail. Section 4 will present the environment configuration and data set, then show and discuss the experimental results. Finally, Section 5 makes a few brief concluding remarks.

2. Related Works

In recent years, deep learning has revolutionized research applications across various fields. In this section, we introduce an image authentication model that is based on Recog-Net and the image detector proposed by Marra et al. [20].

2.1. Recog-Net

Mao et al. proposed an end-to-end image authentication model called Recog-Net [21] based on a deep residual network. The main method determines the authenticity of the image using convolutional neural networks to extract image features by calculating the Mahalanobis distance [22] between the to-be-detected images and labeled images. The Recog-Net focused on how to extract image features with strong characterization ability and accurately describe the feature vector of the image.

The Recog-Net image authentication model based on the deep residual network is divided into three parts in the experiment. The first part is to verify the performance of the feature extraction algorithm to test the accuracy of the authentication model. The performance of the feature-extraction-algorithm-based convolutional neural network is much higher than that of traditional feature extraction. This is because traditional extraction algorithms use (Principal Component Analysis, PCA) and other specific coding such as Bag of Words (BOW), Fisher vector (FV) coding, and Spatial Pyramid Matching (SPM) approaches which strongly rely on prior knowledge. Therefore, it is difficult to capture the essential features of things in complex scenes. On the contrary, the convolutional neural network based on end-to-end learning relies on big data and a space with high-dimensional parameters to gradually synthesize high-level features from the bottom to the top, with an accuracy rate of 94.1%.

The second part is to test the feature versatility of the feature extractor in the Recog-Net network. The overall accuracy of the classification is improved by 3% to 6%. Compared with the classification accuracy of the other networks with feature adjustment layers, the performance of the Recog-Net network is better than other networks, with a classification accuracy of 77.1%.

The third part is to test the accuracy of the image authentication module. A small range of rotation, affine, grayscale change, and other transformations are performed on each picture of the Scene-50 data set to simulate the tampered image. These tampered images and the original Scene-50 data set form a new data set named new Scene-50, and the new Scene-50 data set is used for training. The experimental results finally set the threshold to 2.1 and the accuracy of the public data set in Caltech-101 is 73.6%; using Scene-50, the data set can reach 90.8%. The experimental results show that the authentication model has quite good performance in image authentication.

2.2. Marra et al.'s Method

Marra et al. proposed a detection method of GAN-generated fake-images [20]. The main purpose of this paper is to target the attack method of image-to-image translation and analyze the performance of the image detector after the image is published on the social network.

The experimental data set has a total of 36,000 color images with a size of 256×256, and a total of eight categories. Each category contains real images and fake images. For example, the first group is natural image translation (apple-to-orange, horse-to-zebra), including the original images of apples and oranges, as well as the images of apples and oranges generated through image translation. Marra et al. [20] experimented with three different environments. First, the first environment is to use the original data set (uncompressed) for training and testing. From the average classification accuracy of the eight detectors, both shallow and deep networks have the accuracy rate more than 80%, and from the average classification accuracy of each category, it can be found that both the winter-to-summer translation image and the satellite image obtained through the generation have almost no traces of visual forgery, so the average classification accuracy is poor. In the second

environment, the original data set is used for training, but the compressed data are used for testing. The compression used in the experiment is similar to Twitter's Quality Factor (QF). From the average accuracy, the deep level network is significantly better than the shallow layer, which also shows that the detector of the deep network not only relies on small patterns in the image, but also depends on other features that can be compressed. The third environment is to use the compressed image for training and testing. After testing, it can be found that some features are lost during the compression process, which makes the detector unable to correctly distinguish the image, but the classification accuracy of the deep network detector (XceptionNet) still is 89.03%, maintaining good robustness.

The Recog-Net image authentication model [22] has the advantages of deep residual network, end-to-end model, and high accuracy, but it requires a large amount of training data, computing resources, and there may be some challenges in terms of robustness in abnormal situations. Marra et al.'s image detectors [20] have the advantages of multiple convolutional neural networks, image-to-image translation, and multiple convolutional layers, but require a large quantity of training data and computing resources, and may be at risk of overfitting and parameters. In the next section, we will propose a deep-learning-based architecture that does not require any embedded codes, can directly perform image authentication through the received image, and can indicate whether the image has been tampered with and restore the tampered image blocks.

3. Proposed Method

As image processing technology advances, digital images can be easily duplicated and manipulated, leading to unauthorized tampering and theft of images. Moreover, when digital images are utilized as evidence for significant events, malicious tampering can result in unforeseeable damages. As such, ensuring the authenticity and integrity of digital images has become crucial. To address this issue, this study proposes an image authentication approach that incorporates a coding model based on a VAE and GAN model, and integrates a classification mechanism for undistorted images. This method can locate tampering and self-recovery tampering areas. This method is mainly to detect the entire image in a block-wise manner and determine whether it is the original image or a forged image that has been tampered with according to the probability value of each block detection, and can remove the tampered image, reconstruct the image, and finally locate the exact location of the tampering. The method in this paper is divided into two stages: the training phase and verification phase.

3.1. Training Phase

As shown in Figure 1, in the training process of our image authentication model, a total of five subnets are the encoder €, generator (G_1), generator (G_2), discriminator (D), and classifier (C). The five subnets will be trained together. First, a grayscale original image (I) (size $W \times H$) is divided into blocks (x) (size $m \times m$). Encoder (E) will use $(W \times H)/(\mathrm{m} \times \mathrm{m})$ blocks and label (c) cut from the original image as input. The encoder (E) converts the image block into two common statistical distribution parameters: mean (μ) and standard deviation (σ). The two parameters μ and σ can represent a normal distribution in the latent space of image features. The normal distribution is re-sampled to obtain the hidden vector representing the image feature as the output of the encoder (E). Then, the generator (G_1) uses the hidden vector converted by the image block in the encoder © and the corresponding lab© (c) of the image block as input to reconstruct the original image block (x_r). At the same time, the generator (G_2) also uses a ©el (c) and a normally distributed noise (z), randomly sampling as the input of G_2, and the generated fake image block (x_f), the purpose of which is to improve the generation. The quality of the image generated by the generator (G_1, G_2) accelerates the convergence of the model. The discriminator (D) uses the probability score obtained by distinguishing the original image block (x) from the generated block (x_r, x_f), gives feedback to the generator (G_1, G_2), and fine-tunes the generator (G_1, G_2). Regarding

the weight of the parameters, the discriminator (D) repeatedly distinguishes image blocks and fine-tunes the weight of the parameters of the discriminator.

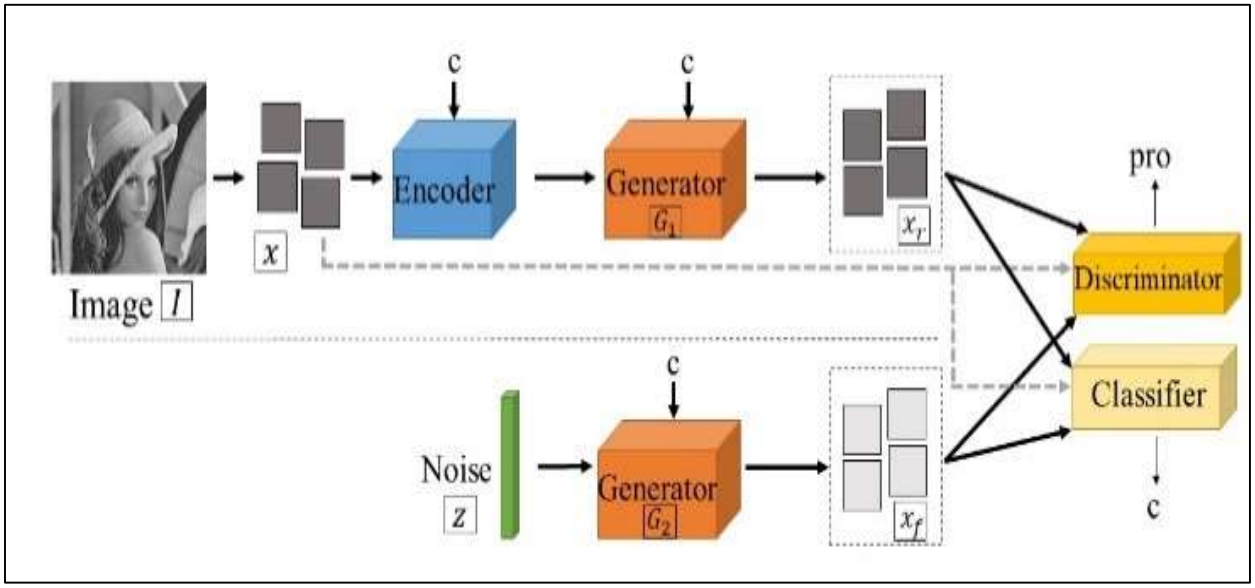

Figure 1. VAE-bGANs image authentication model training flowchart of this research.

Next, we will explain in detail the loss function and architecture defined by the generator (G_1), generator (G_2), discriminator (D), and the newly added encoder (E) and classifier (C).

The generator's loss function L_G consists of two parts, L_{G1} and L_{G2}, as shown in the following Equations (1) and (2). The goal is to minimize the loss function L_G (Equation (3)). Among them, D represents the discriminator, x_f represents the fake image block generated by the generator (G_2), x_r represents the original image after the encoder (E) is converted to the latent space, and the hidden vector is sampled and then input to the resulting reconstructed block of the generator (G_1). We hope that the reconstructed image generated by the generator (G_1) is discriminated by the discriminator as close to 1 as possible, and the forgery image generated by the generator (G_2) is discriminated by the discriminator as close to 1 as possible.

The symbols C represents the classifier and c represents the category label, where the category label indicates the category to which the image block belongs. The "classifier" acts as a "locator", which can be used to identify whether a block number has been tampered with. The closer the probability of x_f through the classifier to its category is, the better. The closer the probability of x_r through the classifier to its category is, the better.

$$L_{G1} = -LogD(x_r) - LogC(c|x_r) \tag{1}$$

$$L_{G2} = -LogD(x_f) - LogC(c|x_f) \tag{2}$$

$$L_G = L_{G1} + L_{G2} \tag{3}$$

The loss of the discriminator consists of three parts in the following Equation (4), and the goal is to minimize the loss function L_D. Among them, x represents the original image block, x_f represents the fake image block generated by the generator (G_2), x_r represent the original image after being converted to the latent space by the encoder (E), and the hidden vector is sampled and then input to the reconstructed block obtained by the generator (G_1). x_r is the same as the above, and we hope that the discriminator discriminates the real image block as close to 1 as possible. The fake image block and the reconstructed image block

should be as close to 0 as possible, using 1 to subtract the judgment result to achieve the goal of minimizing the loss function L_D.

$$L_D = -LogD(x) - Log\left(1 - D\left(x_f\right)\right) - Log(1 - D(x_r)) \tag{4}$$

The loss function L_E of the newly added encoder has two parts, as shown in the following Equation (5). The goal is to minimize this function L_E. The first part is that we hope that the error between the real image block x and the reconstructed image block x_r are as small as possible. The second part is KL divergence loss (Kullback–Leibler divergence), $q_\varnothing(z)$ represents the distribution of the original image mapped to the latent space by the encoder, $p_\theta(z)$ represents the target distribution. We hope that the two distributions are as close as possible, that is, the smaller the error, the better.

$$L_E = \left(\frac{1}{n}\sum|x - x_r|\right) + \int q_\varnothing(z) log\frac{q_\varnothing(z)}{p_\theta(z)}dz \tag{5}$$

The loss of the classifier has three parts in Equation (6), and the goal is to minimize this loss function L_C, where c represents the category of the image block, and each image block has its own category label.

The function of the classifier is very similar to the above discriminator, and the difference is that the discriminator is used to discriminate whether the image block generated by the generator is similar to the original real image block. The classifier is to measure whether the model classifies the image block into the category to which it belongs. Therefore, the first part in Equation (6) is that each real image block x has its own category label c. The higher the probability, the better. The design of the loss function of the second part and the third part is similar to that of the discriminator. In the second part, the higher the probability that x_f belongs to its own category, the better, so the result of the discrimination needs to be subtracted by 1. Similarly, in the third part, the probability that x_r is closer to its own category through the classifier, the better, and the result of the discrimination should be subtracted by 1 to achieve the purpose of minimizing the loss function.

$$L_C = -LogC(c|x) - Log\left(1 - C\left(c\middle|x_f\right)\right) - Log(1 - C(c|x_r)) \tag{6}$$

In the end, our total loss is the sum of the losses of the generator, discriminator, encoder, and classifier, as shown in the following Equation (7).

$$L_{total} = L_G + L_D + L_E + L_C \tag{7}$$

The structures of the five networks are shown in Figures 2–5. The symbols in each structure will be introduced below. In Figure 2, the orange arrow symbol indicates the use of convolutional layer (Conv), batch normalization (Batch_norm), and the activation function of LeakyRelu, and the blue arrow symbol indicates the use of average pooling (Avg_pool) and fully connected layer (Fully_connected). In Figure 3, the blue arrow symbol indicates the activation function using the fully connected layer (Fully_connected), batch normalization (Batch_norm), and LeakyRelu, and the orange arrow symbol indicates the activation using the deconvolution layer (DeConv), batch normalization (Batch_norm), and LeakyRelu function. In Figures 4 and 5, the orange arrow symbol indicates the activation function using convolutional layer (Conv), batch normalization (Batch_norm), and Relu, and the blue arrow symbol indicates the use of average pooling (Avg_pool) and fully connected layer (Fully_connected).

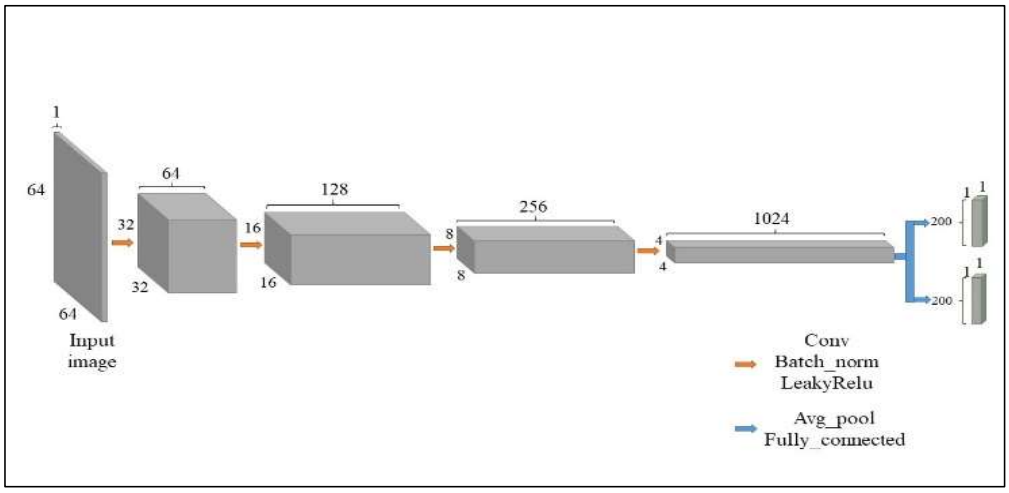

Figure 2. Encoder (E) network architecture of the VAE-bGANs image authentication model of this research.

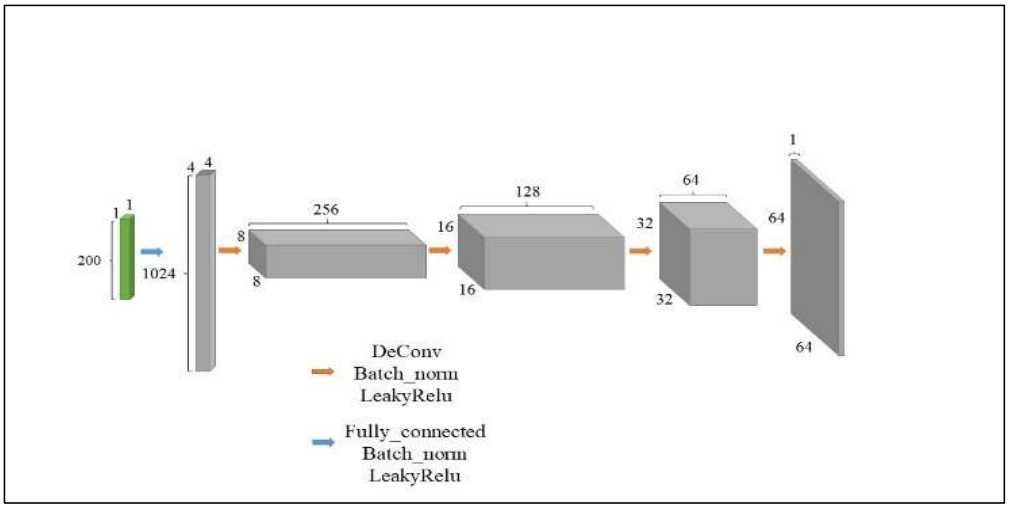

Figure 3. The network architecture of the generator (G_1) and generator (G_2) of the VAE-bGANs image authentication model in this study.

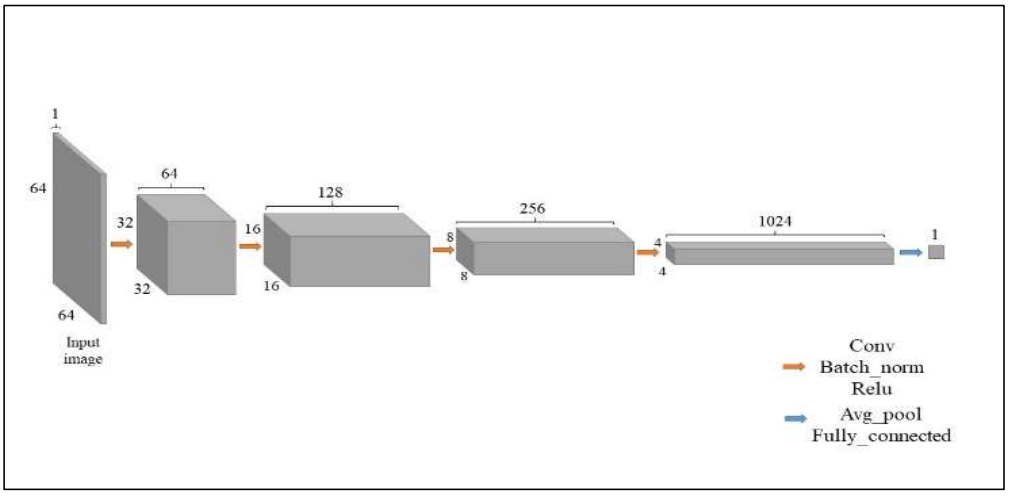

Figure 4. The network architecture of the discriminator (D) of the VAE-bGANs image authentication model of this research.

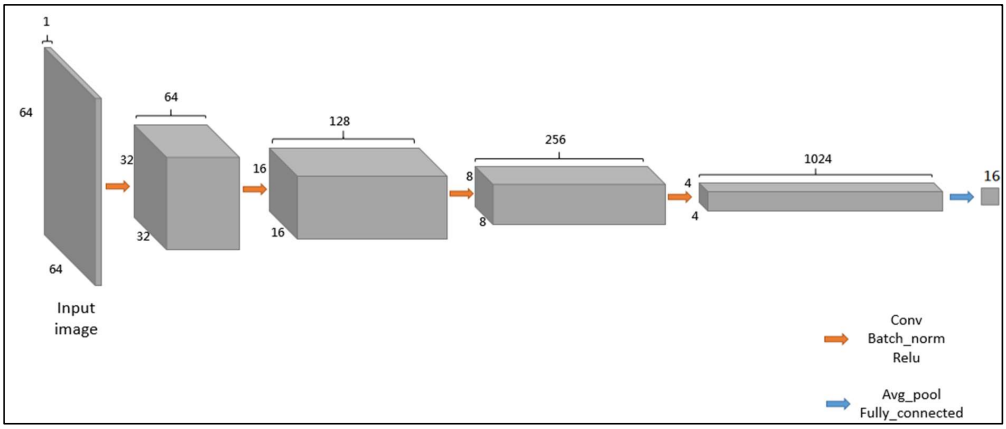

Figure 5. Classifier (C) network architecture of the VAE-bGANs image authentication model in this research.

It is worth noting that the output of the encoder (E) is two 200-dimensional vectors, which are the mean value μ and the standard deviation σ, respectively, which are sampled by Equation (8) to obtain a 200-dimensional vector as the input of the generator. z represents the noise sampled from a normal distribution with a mean of 0 and a standard deviation of 1.

$$y = \mu + \sqrt{e^{\sigma}} \times z \tag{8}$$

3.2. Verification Phase

The overall verification process is shown in Figure 6. We will generate four different maps as tools for subsequent authentication images—namely, the probability map, the residual map, the fusion map, and the final binary map—and the generation of each graph will be described in detail below.

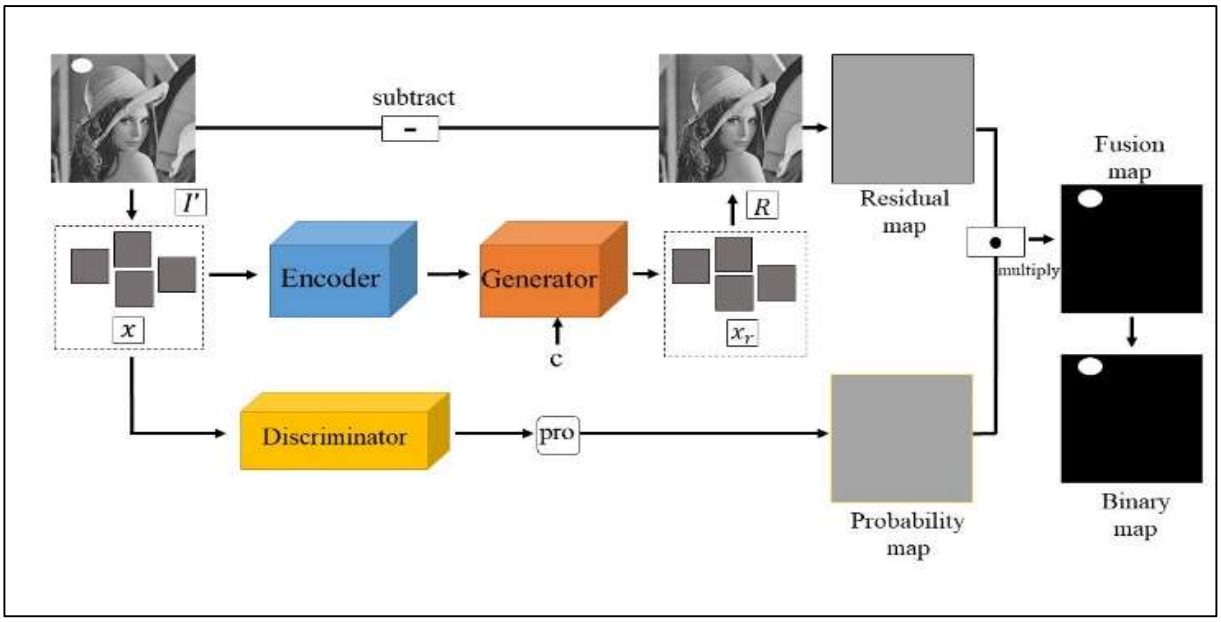

Figure 6. The verification flow chart of the VAE-bGANs image authentication model.

Taking the image that may be tampered with as the input image I', cutting it into blocks x and entering the discriminator. We can obtain the probability P_x that each image block may be tampered with, and reorganize and resize the P_x of each block to obtain the probability map.

Then, each block x containing the classification label is input to the encoder, and then input to the generator to obtain a reconstructed block x_r, and each reconstructed block is reorganized and resized to obtain a reconstructed image R. Before calculating the residual map, we will first calculate the error value err. The err value refers to the pixel subtraction of the reconstructed image R and the input tampered image I', as in Equation (9). Then, the reconstructed image R is converted into a standard reconstructed image R' as in Equation (10).

$$err = \left|I' - R\right|, \; if \; I' \neq R \tag{9}$$

$$R' = \mu(I') + (R - \mu(R))\frac{\sigma(R)}{\sigma(I')} \tag{10}$$

The main purpose of standardized conversion to reconstruct the image is to avoid excessive errors in the subsequent calculation of the residual map. As shown in Equation (11), the input image I' is subtracted from the reconstructed map R' and then multiplied by the error err element by element to obtain a residual map. The symbol "$\cdot$" means multiply element by element.

$$Residual \; map = \left|I' - R'\right| \cdot err \tag{11}$$

Finally, we combine the probability map and the residual map to obtain the fusion map, as in Equation (12). In order to clearly mark the tampered area, the fusion map undergoes binarization conversion to obtain the binary map. The tampered area is located eventually, the tampered area is marked as white, and the normal area is marked as black.

$$Fusion \; map = |1 - (Probability \; map)| \cdot (Residual \; map) \tag{12}$$

4. Experiment Results and Discussion

In this section, we will introduce the experimental environment configuration, data sets, evaluation indicators, and the details and effectiveness of experiments in different situations.

4.1. Experimental Environment and Data Set

Our experimental environment configuration was to use Python 3.6 and Tensorflow framework for development. The experiment was conducted on a Windows 10–64 bit system with 32 GB RAM, Intel (R) Core (TM) i7-9700K CPU, NVIDIA GeForce RTX 2080 Ti, and CUDA 10. The data set comes from the USC-SIPI [23] public standard grayscale library. Figure 7a,b present the most commonly used test images Lena and Baboon, respectively. We randomly take out a Clock from the grayscale image of the USC-SIPI [23] as shown in Figure 7c.

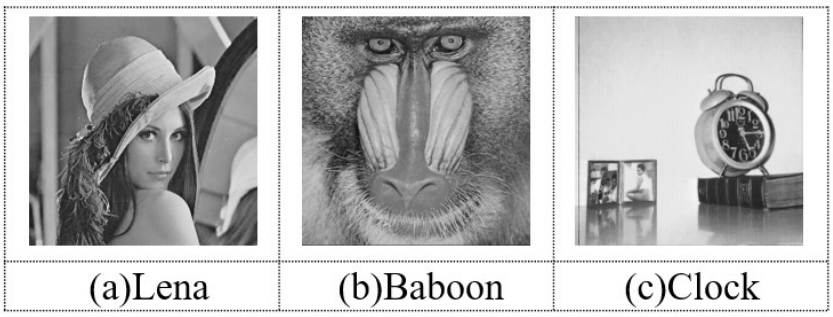

Figure 7. Part of the image of USC-SIPI Image Dataset.

The mechanism we designed is to train a dedicated VAE-bGANs model for each image to be authenticated, enabling it to authenticate that specific image effectively. Each image to be authenticated will be segmented into 64x64-sized image blocks. We further apply image processing techniques such as horizontal or vertical flipping, rotation, scaling, translation, and adding random noise to each image block, thereby increasing to form a dataset with

256 images, each of size 64 × 64. In this dataset, each image block is associated with a corresponding class label. This dataset is then split into training and validation sets in an 8:2 ratio. In our mechanism, we first train the parameters of the discriminator and then proceed to train the parameters of the generator.

4.2. Evaluation Index

The experiment in the proposed VAE-bGANs image authentication will use the following two indicators: peak signal-to-noise ratio (PSNR) and structural similarity (SSIM).

The peak signal-to-noise ratio (PSNR) is an index to calculate the difference between two images through the mean square error (MSE). The typical peak signal-to-noise ratio is between 20 dB and 40 dB. The higher the value, the smaller the error between the two images. The structural similarity index (SSIM) is a method of calculating the structural similarity between two images, and comparing the brightness, contrast, and structure of the images. SSIM will output a value between 0 and 1. The closer the value is to 1, the more similar the two images are.

4.3. Experiments to Construct a Model with the Clock Image

Taking the "Clock" grayscale image in Figure 8a as an example, in order to enable the discriminator to provide appropriate probability values for real and tampered image blocks, we first train the discriminator with the following relevant parameter settings: learning rate of 2.0×10^{-4}, optimizer as Adam, batch size of 1, and training epochs ranging from 10 to 20 (epochs = 10, 11, 12, ..., 20). Next, we will also divide the "Clock" image with a graffito of "NCHU" letters into 16 equal-sized blocks. Among them, the top-left 3 blocks are manipulated regions, and the other 13 blocks are normal and unmanipulated regions (Figure 8b). Each block is attached a class label as shown in Figure 8c.

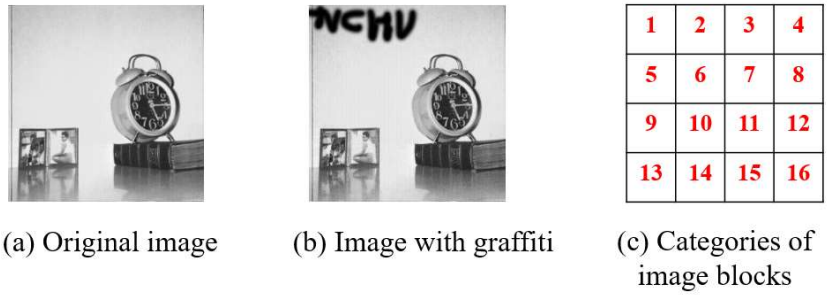

Figure 8. Original Clock image, the Clock image with a graffito of "NCHU" letters, and block categories.

In the training phase, the "Clock" grayscale image of Figure 8a is divided into 16 image blocks with a size of 64 × 64. Each block is given a label, a total of 16 class labels, as shown in Figure 8c. In the graffiti Clock image, three blocks are tampered, which are blocks 1, 2, and 3.

During the training phase, the experimental design first aims to generate detection probability values for each block from the discriminator. For manipulated blocks, their probability values should be close to 0, while for normal blocks, their probability values should be close to 1. During the training of the discriminator, as the number of training epochs gradually increased from 10 to 14, the probability values for manipulated blocks decreased significantly and approached 0, while the values for normal blocks gradually approached 1. However, when the training epochs were increased from 15 to 20, we noticed that some probability values for unmanipulated blocks did not continuously approach 1; instead, they slightly deviated from 1. Accordingly, we decided to select epoch = 14 as the final number of training epochs for the discriminator when determining whether a grayscale image of the "Clock" has been manipulated or not. This decision was made specifically for the clock image shown in Figure 8a. Next, with the relevant parameters of the discriminator frozen, we proceed to train the generator. Figure 9 shows the value of each loss function at different epochs. In addition to observing the change in the loss

function, we will also observe the blocks generated by the model, as shown in Figure 10, so we take the training result of epoch 120 as our final model. For training the overall model, the corresponding categories and image blocks are expanded to 256 blocks as training data. Related parameters include learning rate 2.0×10^{-4}, optimizer Adam, batch size 1, and epoch set to 120.

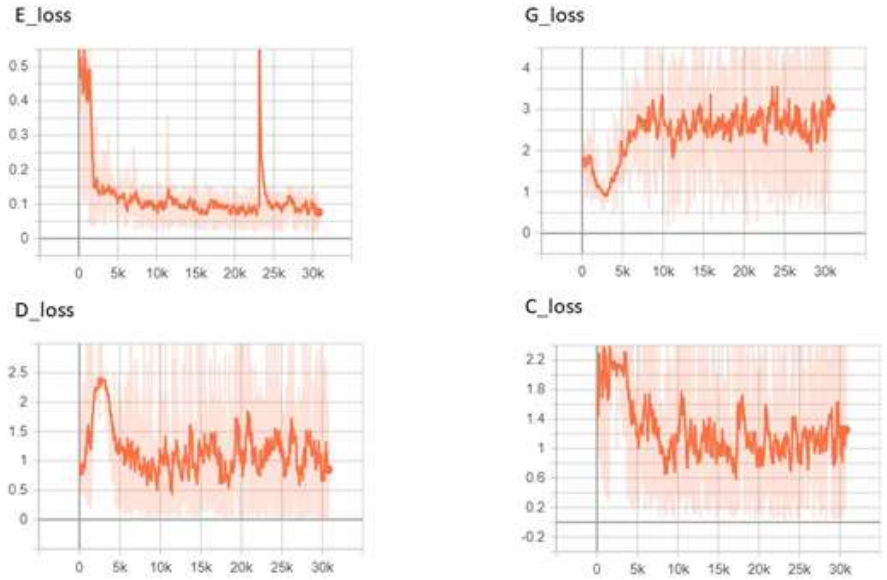

Figure 9. The loss functions of encoder (E), generator (G), discriminator (D), and classifier (C).

Epoch	1	15	30	65	120
Reconstruct patch					

Figure 10. The blocks generated by our model.

Through the VAE-bGANs model for the grayscale image of the "Clock", the result obtained is shown in Figure 11. It can be seen that the reconstructed map (Figure 11b) visually restores the original image as shown in Figure 11a. The probability map (Figure 11c) is generated from the discriminator's probability values, where the tampered areas are shown towards white, and normal areas towards black for better visual distinction. In addition to the graffiti letters appearing in white in the residual map, some areas also appear white sporadically as shown in Figure 11d. The reason is that the generated reconstructed image cannot completely comply with the original image even after standardized conversion. So by combining the operation of the probability map, the final fusion map (Figure 11e) can clearly locate the graffiti regions. The binary map (Figure 11f) generated for visual distinction helps locate the manipulated areas. In summary, the input image is trained through the model, and after the verification phase, each map is produced as the basis for authentication to complete the entire process of image authentication. The results also show the effectiveness of our model in image detection, restoration, and tampering and positioning.

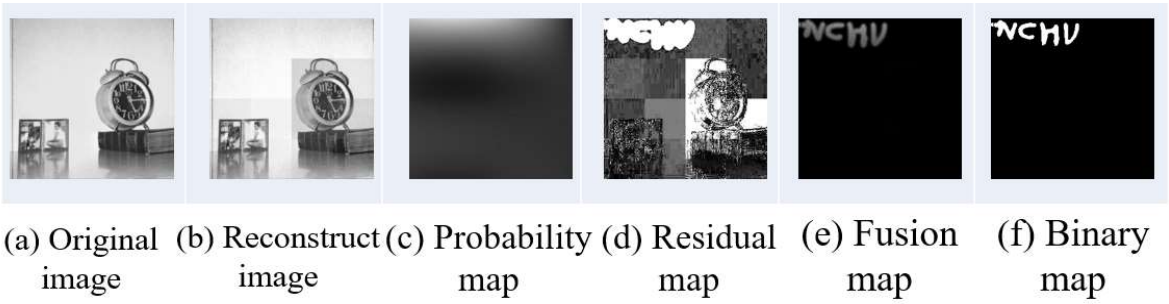

Figure 11. Verification phase result.

4.4. Experiments Using Different Images to Test Multiple Attacks

The proposed VAE-bGANs method uses the Lena and Baboon grayscale images that are often used in image authentication to test the effectiveness of the model in the case of different image attacks. The main attacks are as follows: digging blocks on the image, superimposing patterns on the image (flowers are used here), adding text on the image, image synthesis, image cropping, and merging of two images in half. Experiment with our authentication model. Figure 12 shows the main image attacks used by Lena, and Figure 13 shows the main image attacks used by Baboon.

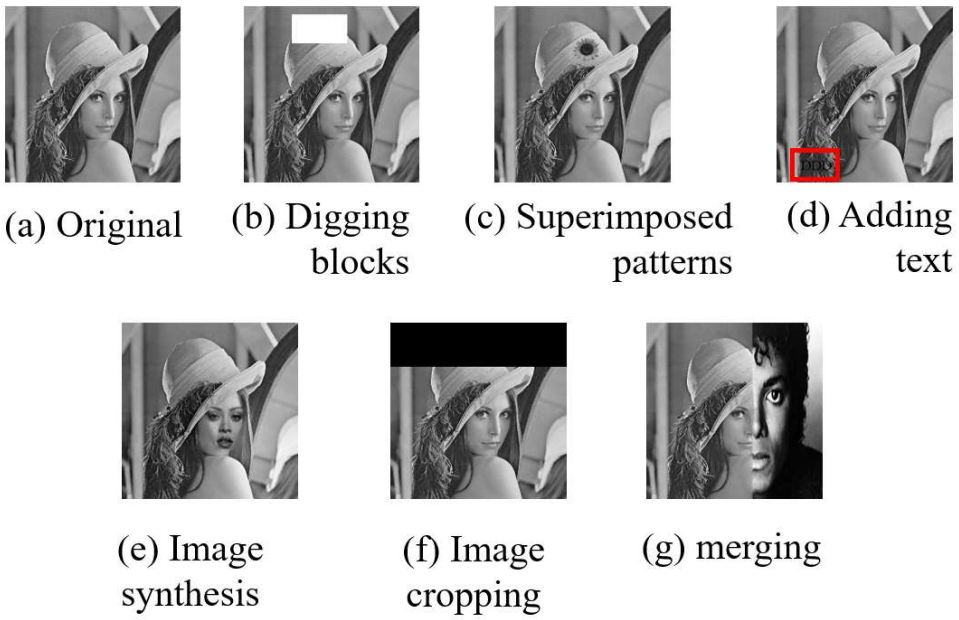

Figure 12. Image attacks mainly used by "Lena" image.

The experimental results of the Lena image are shown below. Figure 14 shows the results of digging blocks on the image. Figure 15 shows the results of superimposing patterns on the image (flowers are used here). Figure 16 shows the results of adding text on the image. Figure 17 shows the results of image synthesis. Figure 18 shows the results of image cropping. Figure 19 shows the results of merging of two images in half. We can see the reconstructed maps in Figures 14b, 15b, 16b, 17b, 18b and 19b. Our model can successfully restore the appearance of the original image. Then, we can directly observe the part of binary maps in Figures 14f, 15f, 16f, 17f, 18f and 19f.

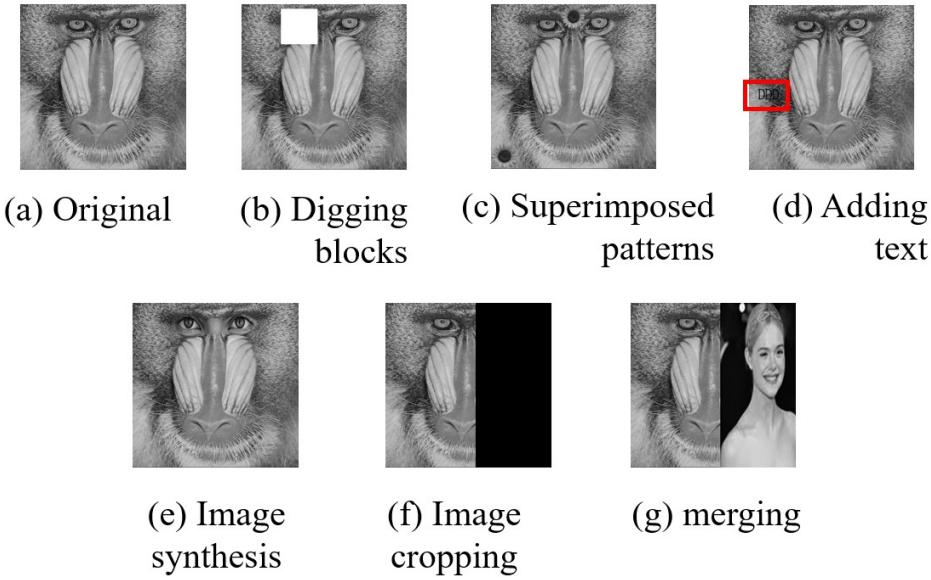

Figure 13. Image attacks mainly used by "Baboon" image.

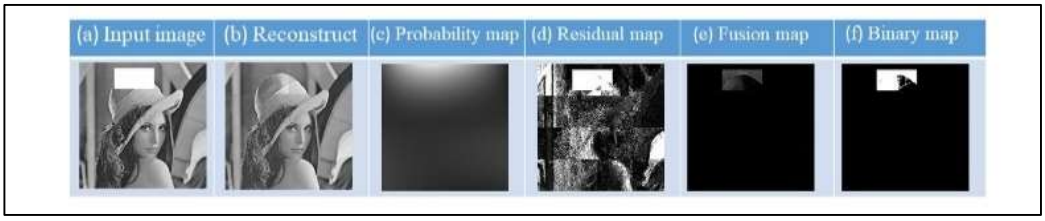

Figure 14. The experimental results of digging blocks on the image "Lena".

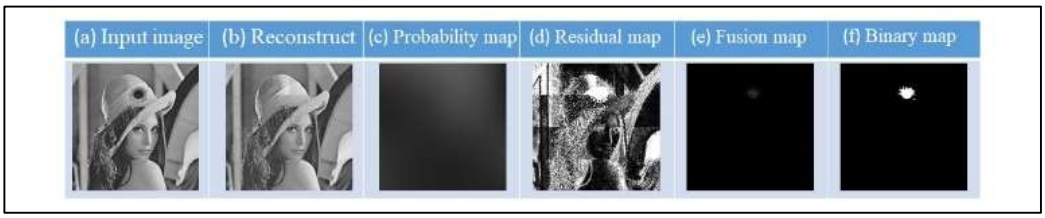

Figure 15. The experimental results of superimposing patterns on the image "Lena".

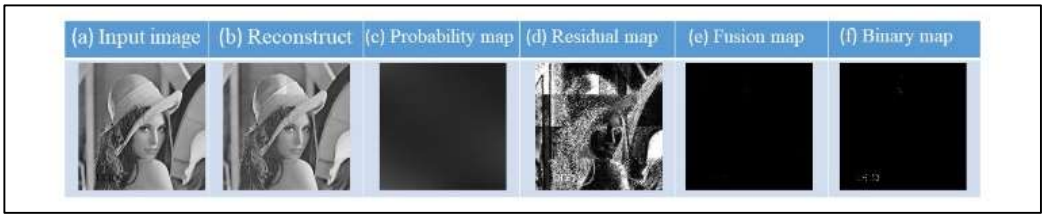

Figure 16. The experimental results of adding text on the image "Lena".

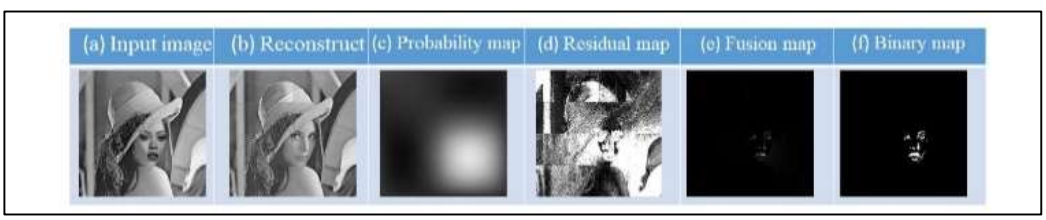

Figure 17. The experimental results of image synthesis using "Lena".

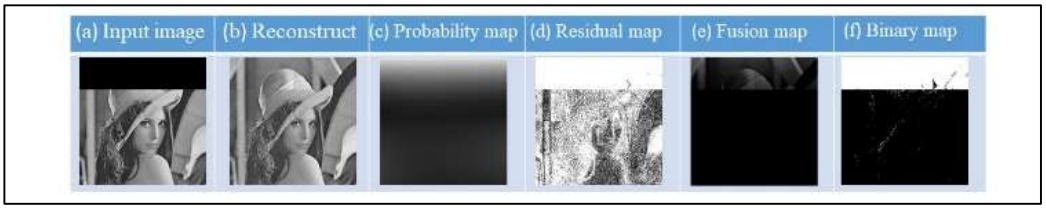

Figure 18. The experimental results of image cropping using "Lena".

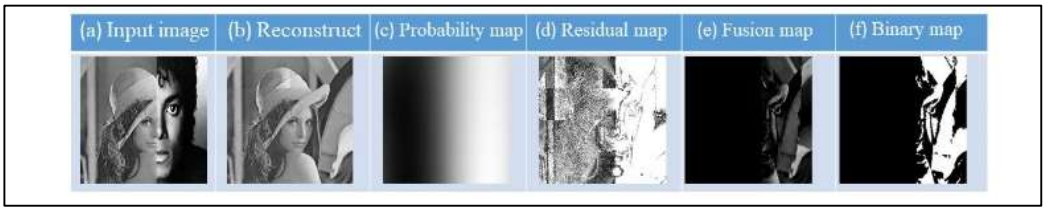

Figure 19. The experimental results of merging of two images in half using "Lena".

Each map can clearly point out the tampered location, which shows that our proposed authentication model can correctly detect the tampered image under different attack situations and also clearly localize the tampered location.

The experimental results of the Baboon image are shown below. Figures 20–25 respectively show the results of digging blocks on the image, the results of superimposing patterns on the image (flowers are used here), the results of adding text on the image, the results of image synthesis, the results of image cropping, and the results of merging of two images in half. We find that there is a little gap between the original image and the reconstructed map shows in Figures 20f, 21f, 22f, 23f, 24f and 25f. Unlike the Lena image, it can be completely restored, but the probability map is modified. Different judgment values between the tampered area and the normal area, the fusion map (or binary map) we calculated can still mark the tampered area, which does not affect the judgment and tampering effectiveness of our authentication model.

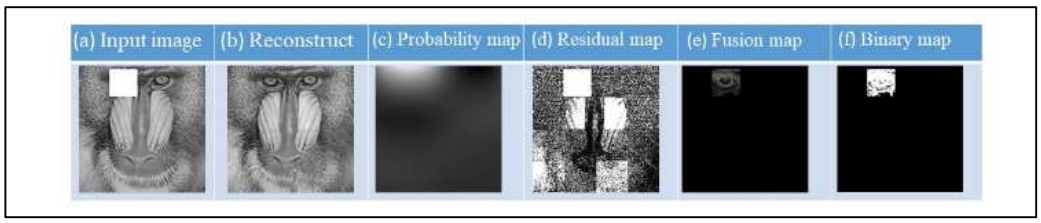

Figure 20. The experimental results of digging blocks on the image "Baboon".

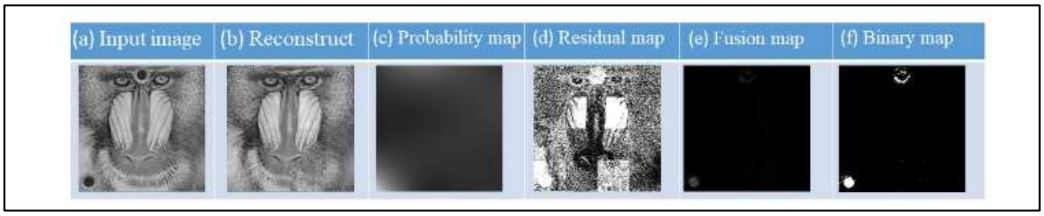

Figure 21. The experimental results of superimposing patterns on the image "Baboon".

As mentioned above, the VAE-bGANs method we proposed is not only used in Clock, Lena and Baboon grayscale images, we also apply it to other images that are often used for image authentication such as Boat, Plane, Man, Pepper, and Sail. We compute PSNR and SSIM values for the reconstructed and original images, respectively. According to the results in Table 1, the average PSNR value and the average SSIM value are 27.2 dB and 0.9, respectively. Both PSNR and SSIM values of the Clock image are within a reasonable range.

The SSIM value of the Clock image is 0.95, and there will be some slight color differences in some areas of the reconstructed image.

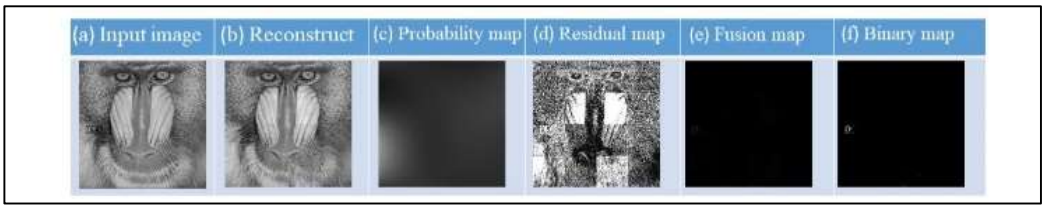

Figure 22. The experimental results of adding text on the image "Baboon".

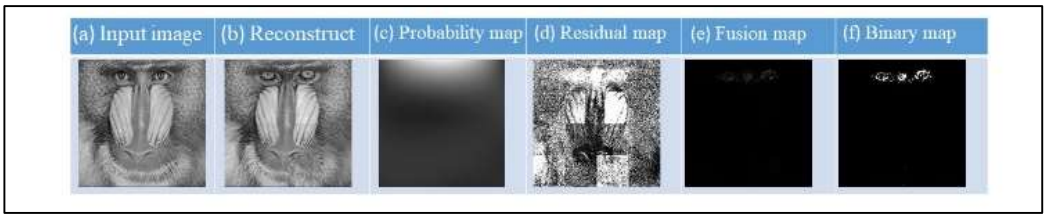

Figure 23. The experimental results of image synthesis using "Baboon".

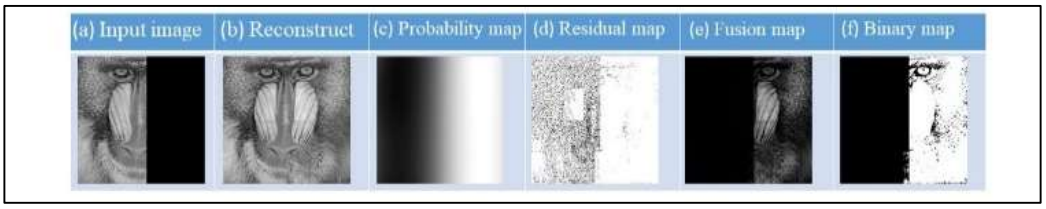

Figure 24. The experimental results of image cropping using "Baboon".

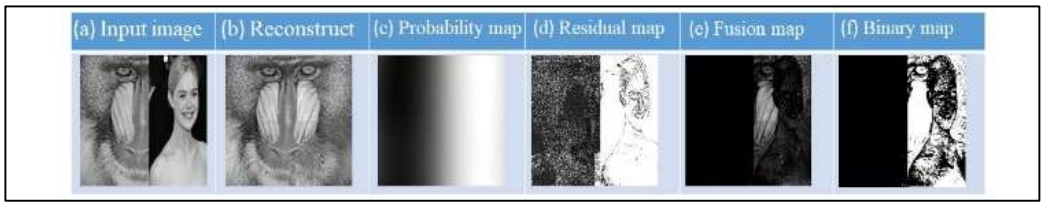

Figure 25. The experimental results of merging of two images in half using "Baboon".

Table 1. PSNR and SSIM values of a variety of standard grayscale images.

Image	PSNR	SSIM
Clock	25.547	0.950
Lena	27.734	0.895
Baboon	25.714	0.857
Boat	27.862	0.913
Plane	27.006	0.925
Man	25.766	0.842
Pepper	29.407	0.924
Sail	28.645	0.923

The main purpose of this paper is to detect the image and localize the tampered area. In addition to identifying the degree of difference between the reconstructed image and the input image based on the proposed method, we also use the probability map

identified by the image detector to distinguish normal blocks from tampered blocks. The difference is enlarged so that the tampered part can be clearly marked, as the basis for image authentication, and the result of image authentication can be credible. It can be seen that our model can be applied to a variety of image tampering attacks, and the feasibility and effectiveness of our method can also be verified on different images.

5. Conclusions

In this article, we propose a new image authentication method based on the VAE coding model and GAN model and add a classification mechanism. The biggest difference between the proposed VAE-bGANs method and the recent deep-learning-based image authentication method is that we use blocks to detect the authenticity of the image, restore the original image through the model, use the result of the image detector to detect the block combined with the operation of the residual map, and finally locate the clear location of the tampering.

The model does not involve the problem of misjudgment because the image detector gives each block the probability of whether it is the original (not tampered), so the value of each block directly reflects the two situations of tampering and not tampering. From the experimental results, we can directly observe the binary maps in Figures 14f, 15f, 16f, 17f, 18f and 19f of the Lena image and Figures 20f, 21f, 22f, 23f, 24f and 25f of the Baboon image. We can see with the naked eye that the fusion graphs (or converted binary graphs) used in the experiment for various attacks such as digging blocks, superimposing patterns, adding text, image synthesis, image cropping, and image merging can clearly indicate where the image has been tampered after being attacked. This shows that the image authentication model we proposed can not only correctly detect attacks in various situations, but also locate the tampered area.

In the future, we summarize two main directions for improvement. The first is the quality of image restoration, just like the problem of image chromatic aberration. Trying to design different image preprocessing methods or improving the model's architecture may ameliorate this problem. Second is the method of locating tampered areas. Trying to change and combine different methods to mark the tampered area may advance the performance.

Author Contributions: Conceptualization, J.-J.S.; Methodology, C.-F.L.; Software, C.-T.Y.; Validation, C.-F.L. and J.-J.S.; Formal analysis, C.-F.L.; Resources, J.-J.S.; Writing—original draft, C.-F.L. and C.-T.Y.; Writing—review & editing, C.-F.L.; Visualization, T.S.; Supervision, C.-F.L.; Project administration, C.-F.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data that support the findings of this study are available from the author, C.-T.Y. upon reasonable request.

Acknowledgments: This research was supported by the National Science and Technology Council, Taiwan R.O.C., under contract number MOST 111-2221-E-324-019-MY2.

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

References

1. Birajdar, G.K.; Mankar, V.H. Digital image forgery detection using passive techniques: A survey. *Digit. Investig.* **2013**, *10*, 226–245. [CrossRef]
2. Ababneh, S.; Ansari, R.; Khokhar, A. Iterative compensation schemes for multimedia content authentication. *J. Vis. Commun. Image Represent.* **2009**, *20*, 303–311. [CrossRef]
3. Ansari, I.A.; Pant, M.; Ahn, C.W. SVD based fragile watermarking scheme for tamper localization and self-recovery. *Int. J. Mach. Learn. Cybern.* **2016**, *7*, 1225–1239. [CrossRef]
4. Umamageswari, A.; Suresh, G.R. Secure medical image communication using ROI based lossless watermarking and novel digital signature. *J. Eng. Res.* **2014**, *2*, 87–108. [CrossRef]
5. Tsai, P.; Hu, Y.; Chang, C. Novel image authentication scheme based on quadtree segmentation. *Imaging Sci. J.* **2005**, *53*, 149–162. [CrossRef]
6. Yang, C.W.; Shen, J.J. Recover the tampered image based on VQ indexing. *Signal Process.* **2010**, *90*, 331–343. [CrossRef]

7. Di, Y.F.; Lee, C.F.; Wang, Z.H.; Chang, C.C.; Li, J. A robust and removable watermarking scheme using Singular Value Decomposition. *KSII Trans. Internet Inf. Syst.* **2016**, *10*, 5831–5848.
8. Singh, D.; Singh, S.K. Effective self-embedding watermarking scheme for image tampered detection and localization with recovery capability. *J. Vis. Commun. Image Represent.* **2016**, *38*, 775–789. [CrossRef]
9. Qin, C.; Ji, P.; Zhang, X.; Dong, J.; Wang, J. Fragile image watermarking with pixel-wise recovery based on overlapping embedding strategy. *Signal Process.* **2017**, *138*, 280–293. [CrossRef]
10. Lee, C.F.; Shen, J.J.; Chen, Z.R.; Agrawal, S. Self-embedding authentication watermarking with effective tampered location detection and high-quality image recovery. *Sensors* **2019**, *19*, 2267. [CrossRef] [PubMed]
11. Lee, C.F.; Shen, J.J.; Hsu, F.W. A Survey of Semi-Fragile Watermarking Authentication. In *Recent Advances in Intelligent Information Hiding and Multimedia Signal Processing*; Smart Innovation, Systems and Technologies; Springer: Cham, Switzerland, 2019; Volume 109, pp. 264–271.
12. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
13. Boroumand, M.; Fridrich, J. Deep learning for detecting processing history of images. *Electron. Imaging* **2018**, *30*, 213-1–213-9. [CrossRef]
14. Tang, H.; Fu, Z.; Ouyang, J.; Song, Y. Image Authentication by Single Target Region Detection. *Artif. Intell. Secur.* **2019**, *11632*, 509–515.
15. Muzaffer, G.; Ulutas, G. A new deep learning-based method to detection of copy-move forgery in digital images. In Proceedings of the Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT), Istanbul, Turkey, 24–26 April 2019; pp. 1–4.
16. Elaskily, M.A.; Elnemr, H.A.; Sedik, A.; Dessouky, M.M.; El Banby, G.M.; Elshakankiry, O.A.; Khalaf, A.A.; Aslan, H.K.; Faragallah, O.S.; El-Samie, F.E.A. A novel deep learning framework for copy-moveforgery detection in images. *Multimed. Tools Appl.* **2020**, *79*, 1–26. [CrossRef]
17. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Proceedings of the 27th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; MIT Press: Cambridge, MA, USA, 2014; Volume 2, pp. 2672–2680.
18. Larsen, A.B.L.; Sønderby, S.K.; Larochelle, H.; Winther, O. Autoencoding beyond pixels using a learned similarity metric. In Proceedings of the of the 33rd International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016; Volume 48, pp. 1558–1566.
19. Razghandi, M.; Zhou, H.; Erol-Kantarci, M.; Turgut, D. Variational Autoencoder Generative Adversarial Network for Synthetic Data Generation in Smart Home. In Proceedings of the 2022 IEEE International Conference on Communications (ICC), Seoul, Republic of Korea, 16–20 May 2022.
20. Marra, F.; Gragnaniello, D.; Cozzolino, D.; Verdoliva, L. Detection of GAN-generated Fake Images over Social Networks. In Proceedings of the 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), Miami, FL, USA, 10–12 April 2018.
21. Mao, J.; Zhong, D.; Hu, Y.; Sheng, W.; Xiao, G.; Qu, Z. An image authentication technology based on depth residual network. *Syst. Sci. Control Eng.* **2018**, *6*, 57–70. [CrossRef]
22. Mahalanobis, A.; Kumar, B.V.; Sims, S.R.F. Distance-classifier correlation filters for multiclass target recognition. *Appl. Opt.* **1996**, *35*, 3127–3133. [CrossRef]
23. University of Southern California; Signal and Image Processing Institute. The USC-SIPI Image Database. Available online: http://sipi.usc.edu/database/ (accessed on 1 July 2023).