applied sciences

MDPI

Article

Self-Supervised Learning for the Distinction between Computer-Graphics Images and Natural Images

Kai Wang

GIPSA-Lab, Grenoble INP, CNRS, Université Grenoble Alpes, 38000 Grenoble, France;
kai.wang@gipsa-lab.grenoble-inp.fr

Abstract: With the increasing visual realism of computer-graphics (CG) images generated by advanced rendering engines, the distinction between CG images and natural images (NIs) has become an important research problem in the image forensics community. Previous research works mainly focused on the conventional supervised learning framework, which usually requires a good quantity of labeled data for training. To our knowledge, we study, for the first time in the literature, the utility of the self-supervised learning mechanism for the forensic classification of CG images and NIs. The idea is to make use of a large number of readily available unlabeled data, along with a self-supervised training procedure on a well-designed pretext task for which labels can be generated in an automatic and convenient way without human manual labeling effort. Differing from existing self-supervised methods, based on pretext tasks targeted at image understanding, or based on contrastive learning, we propose carrying out self-supervised training on a forensics-oriented pretext task of classifying authentic images and their modified versions after applying various manipulations. Experiments and comparisons showed the effectiveness of our method for solving the CG forensics problem under different evaluation scenarios. Our proposed method outperformed existing self-supervised methods in all experiments. It could sometimes achieve comparable, or better, performance. compared with a state-of-the-art fully supervised method under difficult evaluation scenarios with data scarcity and a challenging forensic problem. Our study demonstrates the utility and potential of the self-supervised learning mechanism for image forensics applications.

Keywords: image forensics; self-supervised learning; machine learning; computer-graphics image; neural network; image manipulation; pretext task

Citation: Wang, K. Self-Supervised Learning for the Distinction between Computer-Graphics Images and Natural Images. *Appl. Sci.* **2023**, *13*, 1887. https://doi.org/10.3390/app13031887

Academic Editor: Mostafa Fouda

Received: 27 December 2022
Revised: 26 January 2023
Accepted: 31 January 2023
Published: 1 February 2023

1. Introduction

Nowadays, with the continuous improvement of computational power and image processing/generation algorithms, creating fake or synthetic images is no longer a difficult task, even for non-experts. The content of these images does not reflect what has really happened in the real world, and malicious use of such images may cause serious problems. In order to recover our trust in the authenticity of digital images, researchers have worked for some years on image forensics (i.e., a passive image authentication approach without actively inserting information like watermarks into images), and various forensic methods have been proposed [1–4]. One particular image forensic problem is the detection of computer-graphics (CG) images created by advanced graphics rendering engines [5–7]. CG images can now have a high level of visual realism (*cf.* examples in Figure 1), and more and more researchers are interested in the so-called *CG forensics problem* of distinguishing between CG images (i.e., synthetic images generated by rendering software tools) and natural images (NIs, i.e., authentic images acquired by digital cameras).

Recent CG forensic methods, mostly based on the deep learning paradigm, can achieve very good performance in the classification of CG images and NIs. However, to the best of our knowledge, all existing methods for CG forensics follow the conventional supervised learning framework, which commonly requires a good quantity of labeled samples for

effective training, especially for the training of deep neural networks. However, in practical applications, we often encounter scenarios with scarcity of labeled data; for example, when we want to detect CG images generated by a new rendering engine for which only a very limited number of CG images created by the targeted rendering engine are available. Under such situations, an alternative and more appealing learning mechanism is the so-called *self-supervised learning* approach [8]. This approach appears, in this case, to be a more flexible and more useful technical solution, being less demanding regarding the number of available manually labeled samples. The main idea of self-supervised learning is to effectively make use of the prevalent unlabeled data combined with a training procedure on a self-supervised task (also called a *pretext task*). In such a task, "self-supervised labels" can be generated in an automatic and convenient way, without any intervention of, or effort required from, human manual labeling. Examples of self-supervised pretext tasks are given in the remaining parts of this paper.

Figure 1. Examples of CG images and NIs. The two CG images on the top row were generated respectively by the VRay [9,10] and Corona [11] rendering engines, and the two NIs on the bottom row are, respectively, from the VISION [12] and RAISE [13] datasets. From top left to bottom right, reproduced with permissions respectively from Qusay Abobaker, 2022; from P&M Studio, 2022; from authors of [12], 2017; and from authors of [13], 2015.

To the best of our knowledge, we present in this paper a first study in the literature on employing the self-supervised learning approach to solve the CG forensics problem. Both existing representative self-supervised learning methods and a properly-designed new method are utilized to carry out the forensic classification between computer-graphics images and natural images. More precisely, in our proposed pretext self-supervised task, the objective is to correctly classify authentic images and their various modified versions after applying different kinds of image manipulation operations. Experimental results show that our proposed method, with a new and appropriate pretext task oriented to image forensics, achieved satisfactory and better performance than existing methods. Our contributions are summarized as follows.

- We conduct and report, to our knowledge, a first and comprehensive study on utilizing self-supervised learning for CG forensics.
- We propose a new self-supervised learning method, with an appropriate pretext task, for the distinction between CG images and NIs. Different from existing pretext tasks,

targeted at image content understanding, our proposed pretext task copes better with the considered forensic analysis problem.
- We carry out experiments under different evaluation scenarios for the assessment of our method and comparisons with existing self-supervised schemes. The obtained results showed that our method performed better for the CG forensics problem, leading to more accurate classification of CG images and NIs in different scenarios.

The remainder of this paper is organized as follows. In Section 2 we provide an overview of related work on CG forensics and on self-supervised learning. Section 3 presents the motivation and technical details of the proposed pretext task and our self-supervised method for the discrimination of CG images and NIs. In Section 4, we evaluate our method and compare it with existing methods under different experimental scenarios. We draw conclusions and suggest some future working directions in Section 5.

2. Related Work

2.1. Detection of CG Images

Early methods for the classification of CG images and NIs are mostly based on handcrafted features which can be extracted from either the spatial or a transformed domain of images. In the spatial domain, different kinds of discriminative features were proposed by researchers which were related to physics-motivated information [14], image color [15], statistics of image edges [16], intrinsic image noises [17], hybrid ones [18], etc. Regarding features in a transformed domain, various image transformations were studied from which discriminative features were extracted. Examples include features from the wavelet domain [19,20], the Fourier transform of wavelet coefficients [21], the contourlet and ridgelet transformed domains [22], etc. Handcrafted features are in general easily understandable and explainable; however, the design of such features is quite laborious and the forensic performance often remains sub-optimal.

Since the publication of the landmark paper of AlexNet [23], methods based on the deep learning paradigm [24] have set up new state-of-the-art results for almost every task related to image analysis. There is no exception for the CG forensics problem. Neural networks are trained and discriminative representations are learned in an end-to-end manner for the classification of CG images and NIs, which bypasses the laborious design of handcrafted features and usually leads to better forensic performance. Convolution Neural Networks (CNNs) with specific designs were proposed, with the objective to cope better with the CG forensics problem. Examples of such specific CNN designs include a special statistics pooling layer in [25], cascaded filtering layers in [6], a high-pass pre-processing layer in [26], a two-stream structure in [27] and [28], a specially designed correlation module in [29], a multiscale texture learning module in [30], and a combination of shallow and deep features in [31]. Besides conventional CNNs, other network architectures were exploited for CG forensics, including the Capsule network [32], the recurrent neural network [33], and a recursive network with attention guidance [34]. In general, deep-learning-based methods outperform handcrafted-feature-based methods, mainly due to the powerful learning capacity of deep neural networks.

As far as we know, all existing methods for CG forensics follow the conventional supervised learning framework. In this paper, we present a first study in the literature on CG image detection based on the self-supervised learning mechanism which is in general more flexible and more useful in practical application scenarios.

2.2. Self-Supervised Learning

The supervised deep learning paradigm has achieved strong performances in vast majority of image analysis tasks including the CG forensics problem. However, one well-known limitation of this paradigm is its data-hungry property, as it usually requires a large amount of manually labeled data to carry out an effective training. When we are short of labeled data, the performance may drop. In order to find remedy solutions and pursue the objective of deriving learning methods less demanding in manually labeled

data, a promising and alternative technique is the self-supervised learning mechanism which has become recently a very popular research topic. The research on this mechanism is also in line with the *big objective* of understanding and mimicking the learning activity of human brains which in most cases is not supervised. In self-supervised deep learning, the basic idea is to make better use of a large quantity of readily available unlabeled data for useful representation learning of neural networks by conducting training on a properly-designed self-supervised learning task. In such a task, we use the so-called self-supervised labels which are generated without any human manual intervention. The self-supervised learning mechanism has recently attracted strong research interests and proven to be useful in different fields including computer vision [8], audio analysis [35], medicine and healthcare [36], graph data analytics [37], etc.

Early attempts on self-supervised deep learning for computer vision tasks considered image rotation prediction [38], image colorization [39], relative position prediction of image patches [40,41], etc., as self-supervised learning pretext tasks. The underlying intuition was that in order to do well in these pretext tasks, the network should be able to understand, to some extent, the structure and semantics of images. For instance, in order to accomplish the relative position prediction of patches of a human face image, the neural network should understand that the two human eyes are on top of the nose with one on the left and the other on the right. The representations/features learned in this way would be useful in downstream computer vision tasks like image classification and object detection. Learning methods based on these pretext tasks were valuable attempts, however the performance is rather limited: There is a clear gap between the performance of such self-supervised methods and their supervised counterpart for downstream computer vision tasks.

Recently, another group of self-supervised learning methods have been proposed which are based on the concept of *contrastive learning*. It can be understood that in this group of methods, the pretext task is that learned representations of similar samples should be close to each other while those of dissimilar samples should be distant. This pretext task appears to be more general than the tasks mentioned in the last paragraph and methods based on contrastive learning in general achieve better performances. Technically, similar samples in contrastive learning are often constructed through randomized data augmentation of one unlabeled sample, while dissimilar samples are simply randomly drawn from the pool of available unlabeled data. Different variants of contrastive self-supervised learning frameworks have been proposed. Chen et al. [42] presented the SimCLR (Simple framework for Contrastive Learning of visual Representations) framework which maximized the agreement of projection of two augmented versions of an image sample. The MoCo (Momentum Contrast) framework by He et al. [43], along with an improved version called MoCo-V2 [44], alleviated a common memory bottleneck problem in contrastive learning and improved the learning performance by introducing a specially designed memory bank and a momentum encoder. Grill et al. [45] proposed the BYOL (Bootstrap Your Own Latent) method composed of two sub-networks interacting and learning from each other, which could achieve a good performance without using dissimilar samples. In the method of Barlow Twins of Zbontar et al. [46], an optimization criterion using the cross-correlation matrix of two groups of learned representations was derived for conducting self-supervised learning with similar and dissimilar samples.

The vast majority of existing self-supervised learning methods are proposed for accomplishing typical computer vision downstream tasks. Therefore the design of the learning mechanism, including the adopted pretext task, is naturally oriented for image content understanding. We consider in this study the use of self-supervised learning for an image forensics task, i.e., the discrimination between CG images and NIs, which is somewhat different from computer vision tasks as discussed later in this paper. We propose a new pretext task suitable for the CG forensics problem as presented in the next section.

3. The Proposed Method

3.1. Motivation

In this study on CG forensics, the pretext task for self-supervised learning should be useful for the main task of the classification of CG images and NIs. As mentioned above, existing self-supervised learning methods tend to be in favor of typical computer vision tasks related to image content understanding, i.e., semantic classification and object detection. Image forensic tasks are, in general, different from these tasks [4,47,48], because in image forensics we are more interested in forensic traces that can be telltales of the image generation or processing history. These traces are often not related to image semantics and sometimes even not easily visible. For example, as discussed in the previous section, and noticed by other researchers, the differences between CG images and NIs may reside in subtle traces of image color, noise or relative relationships between pixel values (e.g., discussions presented in [6,29]).

The above observations motivated the design of a new pretext task that can cope better with the problem of distinguishing between CG images and NIs. A natural idea for the design of such a pretext task is to be able to detect different kinds of *deviations from natural images*. The success in achieving this pretext objective would be beneficial for reaching a satisfying classification performance of CG images and NIs, because, intuitively, CG images can be considered a special kind of synthetic images, the generation procedure of which deviates from the acquisition process of natural images. By following this idea and intuition, in our proposed pretext task, we would like to automatically generate, for each NI in the training set, a number of distorted versions by using various image manipulation operations. These distorted versions of NIs represent different variants of deviations from the authentic NIs and would hopefully act as a kind of useful proxy of CG images that the detector would encounter in practical scenarios. With a training step on such a pretext task, the learned representations of neural networks would be sensitive to various deviations from a large number of NIs, which would later be useful for the classification of NIs and CG images.

Natural images are readily and easily available, e.g., those included in the well-known ImageNet dataset [49]. Therefore, the main "ingredient" in the proposed pretext task was the design of manipulation operations to modify NIs. In order to introduce diverse modifications to NIs, we considered *eight different manipulation operations* of different properties, such as the following: color jitter with adjustment of saturation and hue, RGB color rescaling, Gaussian blurring, noise addition, sharpness enhancement, histogram equalization, pixel value quantization, and Gamma correction. The main motivation to choose these manipulations was to cover a large spectrum of diverse changes/deviations from natural images. Natural images have their specific statistics but understanding and modeling natural image statistics is inherently a very difficult problem [50]. Accordingly, modeling deviations from NIs (including the deviations of CG images from NIs) is also a very difficult task. Here we chose to find a workaround and adopted a simple strategy of considering a group of different manipulations to simulate deviations by modifying NIs in different color components (e.g., jitter of color saturation and hue vs. rescaling in RGB space), in different frequency parts (e.g., Gaussian blurring vs. noise addition vs. sharpness enhancement), and with different processing operations on pixel values (e.g., quantization vs. equalization vs. exponential Gamma correction). Hopefully, these various deviations would cover, at least partially, the real deviations of CG images from NIs. Although being intuitive and certainly sub-optimal, our strategy appears to be a rational first attempt on designing a self-supervised pretext task for the forensic analysis of CG images and NIs. Similar to the fact that pretext tasks for image understanding (such as playing the jigsaw puzzles [41] or predicting image rotation [38]) are sub-optimal for downstream computer vision tasks (as reflected by a performance gap between self-supervised and supervised methods), we could not guarantee that the simulated deviations in our method could foresee all possible deviations of CG images from NIs. However, as shown later in our experimental studies, in general, the proposed self-supervised method was able

to accurately detect CG images produced by four advanced rendering engines which remained *unseen* during the self-supervised training. This could be considered evidence of the appropriateness of our designed pretext task, although, of course, we were not able to ensure that our method could work well on any unseen CG images created by other untested rendering algorithms. In addition, we show, in the following, that these manipulations indeed introduced diverse deviations in terms of the simple yet intuitive first- and second-order image statistics.

In Figure 2 we show a natural image from the ImageNet dataset and five modified versions of this NI. We can observe that visually the modified images are quite different from each other, which illustrates the good diversity of introduced modifications. In order to demonstrate that the various modified versions can indeed represent different deviations from the real distributions of NIs, we show in Figure 3 the empirical marginal distribution of pixel values (i.e., embodied by the conventional image pixel histogram) and in Figure 4 an empirical joint-distribution of pixel values (here embodied by the co-occurrences of two horizontally neighboring pixels). The results in these two figures are for the green channel of images and we have similar observations for other channels. It can be seen from Figure 3 that different manipulations tend to introduce different types of distortions, in terms of the pixel value histogram. For example, the quantization operation results in a comb-like histogram, and the Gamma correction operation (here with a factor smaller than 1) stretches the histogram towards big values with the appearance of gaps and peaks in the histogram after manipulation. Similarly, differences in distribution modifications can be observed for the pixel co-occurrences in Figure 4. For instance, Gaussian blurring makes the distribution less dispersed with a thinner concentration band around the diagonal direction. Noise addition has, somewhat, the inverse effect, which, to some extent, expands the band near the diagonal. The diversity of the considered manipulations, as demonstrated in Figures 3 and 4, would be helpful to learn appropriate representations of neural networks for the main task of differentiating between CG images and NIs.

Figure 2. The image on top-left corner is a natural image from the ImageNet dataset [49]. The remaining five images, from left to right and from top to bottom, are five modified versions of the NI, after applying, respectively, color jitter, Gaussian blurring, noise addition, pixel value quantization (separately in RGB channels), and Gamma correction.

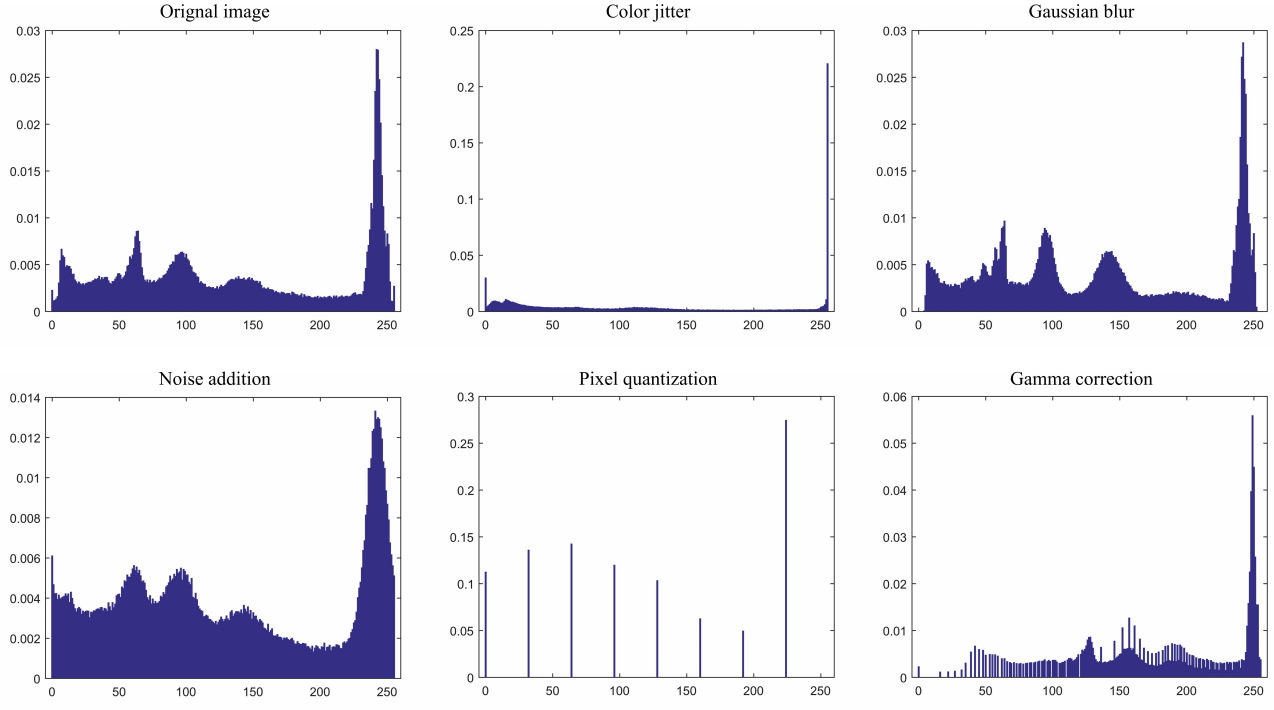

Figure 3. Normalized histogram of pixel values of the green channel of the six images in Figure 2. In each sub-figure, the horizontal axis represents the pixel value (between 0 and 255) and the vertical axis represents normalized occurrence. Please notice that the range of the vertical axis is not always the same for the six sub-figures.

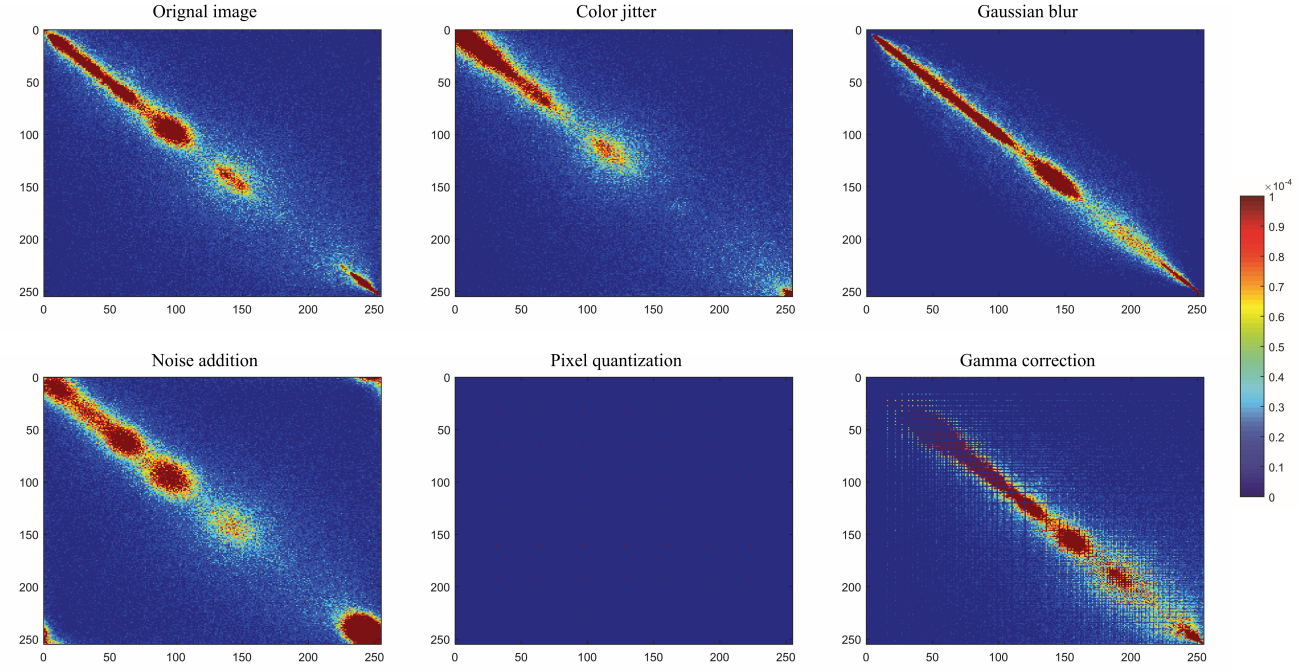

Figure 4. Normalized co-occurrences (shown as heatmaps) of two horizontally neighboring pixels in the green channel of the six images in Figure 2. In each sub-figure, the horizontal and vertical axes represent two pixels' values (between 0 and 255). The color map used to represent normalized co-occurrence values in the six sub-figures is given at the rightmost of the figure. Zoom-in is recommended to well observe the red dots in the sub-figure of the pixel quantization operation.

3.2. Self-Supervised Learning Task

The objective of our method was to first of all carry out self-supervised training on an appropriate pretext task of correctly classifying authentic images and several kinds of processed images obtained after applying different kinds of manipulation operations on authentic images. This allowed us to learn features/representations that would then be utilized to solve the main task of the discrimination of synthetic CG images created by advanced rendering engines and natural images acquired by digital cameras. We would like to point out that both CG images and NIs involved in the main task remained unseen during the self-supervised learning stage. As mentioned in the previous subsection, in our proposed pretext task we considered eight manipulation operations, which are listed in the first column of Table 1. In the second column we present the specifications concerning their parameters, if any. In practice, for applying manipulations on an image, the random parameters, if any, are drawn uniformly from the ranges/set as given in Table 1. For example, in the color jitter operation, two random scaling factors are drawn, respectively, for the scaling of the color saturation and the color hue, while in the RGB color rescaling operation three factors are drawn randomly and independently and then applied, respectively, on the R, G and B channels. As mentioned in the last subsection, we chose the eight manipulations in Table 1 to introduce a large spectrum of diverse modifications to NIs, in order to simulate, to some extent, the deviations of CG images from NIs. The selection of manipulations is rather empirical but as shown in Section 4 this leads to good forensic performances in different scenarios for the discrimination of CG images and NIs. In general, the selected manipulations are believed to be related to the difference between CG images and NIs. For example, several previous methods extracted color features which proved to be effective for detecting CG images [18,21]. Therefore, color jitter, color rescaling or even Gamma correction operation would allow us to learn appropriate representations sensitive to color changes and, thus, be discriminative in distinguishing CG images and NIs. Furthermore, in the literature, researchers have found that CG images and NIs differ in terms of edge statistics [16] and local smoothness [6]; accordingly learned features sensitive to Gaussian blurring, noise addition and sharpness enhancement would be useful for the classification of CG images and NIs. In addition, intuitively and perceptually, CG images appear in general less "colorful" than NIs, probably due to computational or algorithmic limitations of rendering engines. Hence, the pixel value quantization might also be helpful for the considered CG forensic problem. Finally, including other manipulations not that closely related to deviations of CG images (one possible example might be the histogram equalization) would still be helpful to learn representations sensitive to more changes from NIs in general, which would later be more easily adapted to the main task of CG forensics.

Table 1. Manipulation operations considered in the proposed self-supervised method, the specifications about their parameters, and their indices in the multi-class classification pretext task. The implementation of these manipulations was partially based on PyTorch.

Manipulation	Parameter	Index
Color jitter	$FactorSaturation \in [0, 2.0]$, $FactorHue \in [-0.5, 0.5]$	1
RGB color rescaling	$ScalingFactors \in [0, 2.0]$	2
Gaussian blurring	$KernelSize = (3, 3)$, $Sigma \in [0.1, 5.0]$	3
Random Gaussian noise	$Mean = 0$, $StandardDeviation = 20.0$	4
Sharpness enhancement	$EnhanceFactor \in [1.0, 4.0]$	5
Histogram equalization	–	6
Pixel value quantization	$NumberBits \in \{2, 3, 4, 5, 6, 7\}$	7
Gamma correction	$CorrectionFactor \in [0, 2.0]$	8

One may design the pretext task as a binary classification problem between authentic images and modified ones generated by any of the eight manipulations. However, in practice, this binary-classification pretext task leads to limited performance for the main task of CG forensics, probably due to the somewhat over-simple nature of such a binary

task. A more effective, and also more challenging, pretext task is the *multi-class classification task* of authentic images and their eight modified versions obtained after applying the eight manipulations listed in Table 1. The success in this multi-class classification task requires learning more sophisticated and discriminative representations, not only capable of classifying between NIs and their modified versions, but also able to accurately predict the type of manipulation of each modified image. Similar observations were mentioned in [38], where the authors found that the multi-class classification of a number of rotation angles of an image was a more effective pretext task than the simple binary classification between original un-rotated images and any rotated ones.

Therefore, in our self-supervised pretext task there were in total 9 classes of training samples, i.e., the NI (with ground-truth label 0) and modified images corresponding to the eight kinds of manipulations (with ground-truth label in the set of $\{1, 2, \ldots, 8\}$, as shown in the last column of Table 1). Training images of different classes for the 9-class pretext task were generated dynamically. For each image x_i in a batch of NIs drawn from the ImageNet dataset [49] (for the sake of simplicity, and with a little abuse of notation, we omit the batch index), a pseudo-random number y_i was drawn from the uniform distribution on the set of $\{0, 1, 2, \ldots, 8\}$, and then a training sample $\hat{x}_i$ was constructed for the pretext task as:

$$\hat{x}_i = \begin{cases} x_i & \text{if } y_i = 0, \\ MP_{y_i}(x_i) & \text{if } y_i \in \{1, 2, \ldots, 8\}. \end{cases} \tag{1}$$

In the above equation, $MP_{y_i}(.)$ stands for the manipulation indexed by y_i (as mentioned above, the indices of manipulations are given in the last column of Table 1) with randomly drawn parameters, if any. It can be seen that with the dynamic sample generation procedure of Equation (1) and with a large number of NIs (which was the case in our pretext task training), we could ensure that the considered 9-class classification problem was well balanced. Usually, a balanced problem facilitates the learning and implementation, but it is possible that an unbalanced self-supervised learning problem would be more beneficial for downstream tasks, e.g., during the learning process we could dynamically generate more samples for difficult manipulations with a higher training/validation error. In the future we plan to carry out studies to get better understanding and make improvements regarding this point.

It can be seen that $y_i \in \{0, 1, 2, \ldots, 8\}$ was, in fact, the ground-truth label of the generated sample $\hat{x}_i$. We can also observe that the self-supervised labels y_i were generated automatically and dynamically *without* any effort in terms of human manual labeling. Afterward, the pretext task training was carried out by minimizing the conventional cross-entropy loss of the 9-class classification problem as (again with a little abuse of notation):

$$\mathcal{L} = -\frac{1}{M} \sum_{i=1}^{M} \sum_{l=0}^{L-1} \log \frac{\exp(s_i^l)}{\sum_{l=0}^{L-1} \exp(s_i^l)} \cdot \mathbb{1}\{y_i = l\}, \tag{2}$$

where M represents the number of training samples used for the pretext task training, $L = 9$ meant that what we tried to solve was a 9-class classification problem, s_i^l with $l = 0, 1, 2, \ldots, 8$ are the raw (non-normalized) class scores as computed by the neural network, and $\mathbb{1}\{\cdot\}$ stands for the indicator function with $\mathbb{1}\{\text{False}\} = 0$ and $\mathbb{1}\{\text{True}\} = 1$.

It can be observed that our proposed self-supervised training procedure did not require any human manual labeling effort and did not use any CG images, which ensured that our method was very flexible and useful in practical application scenarios. After the self-supervised training on the pretext task, the obtained neural network could later be used to solve the main task of classifying CG images and NIs. The evaluation protocols for the main task and the obtained results are presented in the next section.

4. Experimental Results

4.1. Implementation Details and Datasets

The proposed self-supervised learning method was implemented with PyTorch and based on the VISSL library [51]. The main task considered in our study was the discrimination between CG images and NIs. For the experimental evaluation of CG forensics performance, we used the four recent datasets collected by Quan et al. [27] (some example images are shown in Figure 1), in which the CG images were generated, respectively, by four advanced rendering engines (i.e., Artlantis [52], Autodesk [53], Corona [11] and VRay [9,10]), while the NIs were from the popular RAISE [13] and VISION [12] datasets. Following Quan et al. [27], hereafter we used, respectively, Artlantis, Autodesk, Corona and VRay to denote the four datasets. We adopted the same dataset splits as those of the original paper [27]: For each of the four datasets, the training set comprised 5040 CG images and 5040 NIs, and the testing set had 360 CG images and 360 NIs. We made use of the well-known ImageNet dataset [49] for the self-supervised pretext task training, and this also ensured that there was no overlapping between NIs used for pretext task training (from ImageNet) and those used for main task evaluation (from RAISE and VISION). ImageNet is a huge dataset, comprising more than 1 million images in its training set. We randomly drew 10% ImageNet training images and obtained around 128 K images which were then used for the training of our pretext task, i.e., the 9-class classification problem presented in Section 3.2. We used the well-known and popular ResNet50 [54] as the backbone neural network. The pretext task training was carried out by using the SGD (Stochastic Gradient Descent) optimizer for 105 epochs, and we set an initial learning rate of 0.1, which was successively divided by 10 at 30, 60, 90 and 100 epochs. This training was conducted on two NVIDIA Quadro GPUs (Graphics Processing Units) and, typically, lasted for about 14 h. Although the performance of the pretext task was not the focus in self-supervised learning [8], in practice we found that our pretext task (a 9-class classification problem) could be solved reasonably well with an accuracy of about 85%. As presented later in this section, in our study we conducted experimental comparisons with several popular and representative self-supervised methods. In order to ensure fair comparisons, the self-supervised training of these existing methods was carried out on the same data, for the same number of epochs and with the same backbone neural network as our pretext task, by using the training scripts provided in the VISSL library [51].

4.2. Linear Classification Evaluation Protocol

One common yet rough way to evaluate self-supervised learning methods is to carry out tests under the so-called linear classification evaluation protocol [8,51]. The learned representations by self-supervised learning were utilized directly by a linear classifier on the main task, which was, in our case, the discrimination between CG images and NIs. More precisely, for a neural network (ResNet50 in our study) trained after self-supervised learning, we replaced the last layer of neural network by an appropriate linear layer and trained the linear layer on the main task of CG forensics, while keeping the other parameters of neural network frozen. This meant that only the new linear layer/classifier was trained on the training samples of the CG forensics datasets of Artlantis, Autodesk, Corona and VRay mentioned above, while the other layers of neural network obtained after self-supervised pre-training were untouched. Table 2 presents the experimental results for this linear classification evaluation protocol, where our proposed method is compared with several popular and representative self-supervised methods: Gidaris et al.'s method [38], based on the image rotation prediction pretext task (denoted by RotNet), Noroozi et al.'s method [41], based on the patch puzzles pretext task (denoted by Jigsaw), and three recent methods, based on contrastive learning (i.e., BarlowTwins [46], SimCLR [42] and MoCo-V2 [44]). We also compared with a neural network pre-trained by using the supervised ImageNet image classification task (denoted by ImageNetClassi). The linear classifier of all the methods was trained on the same data and with the same training parameters. From the results in Table 2 we can observe that the performance, after linear classifier training,

was, in general, satisfying and that our proposed method performed consistently well and better than existing methods for the CG forensics problem. This was probably due to better appropriateness of our designed pretext task for the classification of CG images and NIs, leading to more effective representations learned after self-supervised pre-training.

Table 2. Test accuracy of different methods under the linear classification evaluation protocol. The presented results are classification accuracies on the testing set of the four CG forensics datasets of Artlantis, Autodesk, Corona and VRay.

Method	Artlantis	Autodesk	Corona	VRay
RotNet [38]	82.64%	84.44%	79.44%	83.61%
Jigsaw [41]	84.31%	87.64%	83.19%	86.39%
BarlowTwins [46]	75.14%	80.00%	76.11%	80.83%
SimCLR [42]	86.25%	79.17%	86.94%	87.22%
MoCo-V2 [44]	88.89%	81.53%	88.47%	90.00%
ImageNetClassi	87.08%	90.28%	85.69%	89.58%
Ours	92.50%	94.86%	92.08%	91.81%

4.3. Fine-Tuning Evaluation Protocol

A more practical, and more important, assessment methodology for self-supervised learning methods is the fine-tuning evaluation protocol [8,51]. Different from the linear evaluation protocol in the last subsection, all parameters of neural network obtained after self-supervised pre-training (with a new and suitable classification layer for the main task) are fine-tuned on training samples of the main task. This reflects the real practical utility of learned pre-trained neural networks for solving our target problem of CG forensics. Table 3 presents the obtained results when the fine-tuning was carried out on the whole training set of the four CG forensics datasets (i.e., fine-tuned on 5040 CG images and 5040 NIs). Again, the data and the parameters of fine-tuning were the same for all self-supervised methods. Besides the results of our method and the comparison methods considered in the last subsection, we also provide, in Table 3, the result of a state-of-the-art fully-supervised method called ENet [27]. It can be observed that performances of self-supervised methods, both ours and existing ones, were, in general, largely improved when compared to those of the linear classification protocol. Our method still outperformed existing self-supervised methods. There was a small performance gap between our self-supervised method and the fully-supervised ENet method; this was rather understandable, because ENet is a specially designed neural network, having architecture and training well tailored for solving the CG forensics problem in a fully-supervised setting. Moreover, in order to test the utility of our proposed method in a challenging data scarcity scenario, we also conducted tests under the setting where pre-trained networks were fine-tuned only on 20% of the training set of the four CG forensics datasets (i.e., fine-tuned on 1008 CG images and 1008 NIs), and the obtained results are presented in Table 4. We can observe that test accuracies in this challenging, yet practical, situation decreased, and our proposed method had the least decrease (test accuracy remained higher than 92% on all the four datasets under this challenging data scarcity scenario). Our method again outperformed existing self-supervised methods and the network pre-trained on the ImageNet classification task (denoted by ImageNetClassi). In addition, the performance gap between our method and the fully-supervised ENet became slightly smaller (the average of the difference of performances between the two methods on the four datasets changed from 1.97% in Table 3 to 1.59% in Table 4). This was, in our opinion, mainly due to the effective pre-training with an appropriate pretext task and on a good number of data for our self-supervised method. Please also note that in order to reach the performances shown in Tables 3 and 4, ENet (with default training parameters as suggested in [27]) took longer training epochs than the fine-tuning of our method (i.e., 300 epochs for ENet vs. 105 epochs for ours). If ENet was trained with the same number of 105 epochs as our method, its test accuracy dropped to 93.47%, 90.83%, 91.39% and 94.58%, respectively, for the four datasets. In this case, our

method performed better than ENet on three of the four datasets (i.e., 96.25% vs. 93.47% on Artlantis, 94.31% vs. 90.83% on Autodesk, and 93.06% vs. 91.39% on Corona), while ENet performed better on VRay (92.64% for ours vs. 94.58% for ENet).

Table 3. Test accuracy of different methods under the fine-tuning evaluation protocol where the fine-tuning was conducted on the whole training set of CG forensics datasets. The presented results are classification accuracies on the testing set of the Artlantis, Autodesk, Corona and VRay datasets.

Method	Artlantis	Autodesk	Corona	VRay
RotNet [38]	89.86%	91.53%	89.31%	89.72%
Jigsaw [41]	93.06%	94.17%	92.78%	92.08%
BarlowTwins [46]	81.25%	84.72%	79.72%	80.42%
SimCLR [42]	93.47%	95.00%	91.67%	91.81%
MoCo-V2 [44]	96.11%	95.97%	92.50%	94.17%
ImageNetClassi	92.78%	94.72%	90.97%	91.94%
ENet (supervised) [27]	98.69%	98.44%	98.50%	98.75%
Ours	97.22%	97.08%	96.53%	95.69%

Table 4. Test accuracy of different methods under the fine-tuning evaluation protocol where the fine-tuning was conducted on 20% of the training set of CG forensics datasets. The presented results are classification accuracies on the testing set of the Artlantis, Autodesk, Corona and VRay datasets.

Method	Artlantis	Autodesk	Corona	VRay
RotNet [38]	84.31%	86.94%	83.47%	83.75%
Jigsaw [41]	87.08%	91.25%	85.42%	86.94%
BarlowTwins [46]	73.19%	76.67%	70.83%	74.72%
SimCLR [42]	87.36%	87.92%	85.83%	85.83%
MoCo-V2 [44]	90.97%	92.22%	88.06%	89.44%
ImageNetClassi	88.33%	90.00%	85.69%	86.67%
ENet (supervised) [27]	96.81%	95.28%	94.85%	95.69%
Ours	96.25%	94.31%	93.06%	92.64%

Multi-class CG forensics. Furthermore, we considered an even more challenging CG forensics problem for the fine-tuning evaluation, i.e., the multi-class classification problem, in which the objective was to correctly classify NIs and CG images created by different rendering engines. This problem had good practical utility because it might allow forensic analysts to better know the source of a given image, being able to not only discriminate between NI and CG image, but also identify the source rendering engine for a CG image. In our study, this was a 5-class classification problem with the five classes being NIs and four kinds of CG images created, respectively, by Artlantis, Autodesk, Corona and VRay. The dataset of this 5-class classification problem was obtained in a simple way from the four datasets used above: CG images were directly from the corresponding dataset, while NIs were randomly drawn from all the NIs in the four datasets. This was done separately for the training and testing sets and in the end we had a training set of 25,200 images (5040 images for each of the five classes) and a testing set of 1800 images (360 images for each class). Similar to the experiments described above, we still considered two versions of fine-tuning, respectively, on the whole training set and on 20% of training set. Table 5 presents the obtained results. All self-supervised methods had the same data and parameters for the fine-tuning. We slightly modified the last classification layer of ENet so that it could cope with the considered 5-class classification problem. The performances of this adapted ENet are also reported in Table 5. For fair comparisons, all the methods had the same number of 105 epochs for fine-tuning or training. It can be noticed from Table 5 that for this challenging CG forensics problem (a random guess gave 20% accuracy), existing self-supervised methods resulted in rather limited performances, especially for the scenario of fine-tuning on only 20% training data. Our self-supervised method, in this case, performed much better than the existing methods, with a large margin compared

to the second-best self-supervised method of MoCo-V2, i.e., 83.22% (ours) vs. 72.06% (MoCo-V2) and 71.50% (ours) vs. 57.33% (MoCo-V2), respectively, for the two fine-tuning scenarios. In addition, although the supervised and adapted ENet clearly outperformed our self-supervised method when the fine-tuning was carried out on the whole training set, the performances were comparable under the fine-tuning on 20% of data (ours being even slightly better, i.e., 71.50% for ours vs. 71.06% for ENet). This might be an indication of the importance and advantage of a properly designed forensics-oriented pretext task for a self-supervised learning method, in particular under a data scarcity scenario with a very challenging forensic problem.

Table 5. Test accuracy of different methods under the fine-tuning evaluation protocol for the 5-class classification problem (the 5 classes being NI, CG of Artlantis, CG of Autodesk, CG of Corona, and CG of VRay). The presented results are classification accuracies on the testing set of a dataset properly constructed from the datasets of Artlantis, Autodesk, Corona and VRay (*cf.* the main text for details).

Method	Fine-Tune on Whole Train Set	Fine-Tune on 20% Train Set
RotNet [38]	61.39%	50.11%
Jigsaw [41]	65.61%	53.22%
BarlowTwins [46]	44.94%	33.22%
SimCLR [42]	66.50%	51.39%
MoCo-V2 [44]	72.06%	57.33%
ImageNetClassi	70.28%	54.61%
ENet adapted (supervised)	90.61%	71.06%
Ours	83.22%	71.50%

4.4. Additional Experiments

In this subsection, we present additional results related to the design and evaluation of our proposed self-supervised method.

4.4.1. Impact of Manipulations

Experiments were conducted to study the impact of the pretext task design in our method. As presented in Section 3.2, we adopted a pretext task of 9-class classification of NIs and their modified versions, obtained by applying eight different manipulations. Here we compared with two other versions of pretext task, respectively, with 6 classes and 23 classes. In the version with 6 classes, we removed from the 9-class version the three manipulations of histogram equalization, pixel value quantization and Gamma correction. To get the version with 23 classes, we expanded the 9-class version with more variants for some manipulations. For example, in the 23-class version the Gamma correction manipulation had two ranges for the correction factor, i.e., between 0 and 1 (resulting in a brighter image after correction) and between 1 and 2 (resulting in a darker image). For the noise addition manipulation, we also considered the three variants with added noise only in one of the three color channels. Our purpose was to have a first and rough idea about the impact of the number of manipulations and their subdivisions with more parameters/settings on the performance of the main task of CG forensics. We tested the performance of these three pretext tasks (i.e., 6-class, 9-class and 23-class versions) under the linear and the fine-tuning (on 20% training set) evaluation protocols. The performance comparisons are illustrated in Figure 5. We can observe that, in general, the 9-class version had consistently better performance than the 6-class version (except on Corona under the linear evaluation protocol where the performance was equal), while the 23-class version did not clearly perform better than the 9-class version and, sometimes, the performance of 23-class pretext task could be lower than that of the 9-class task (e.g., on Autodesk and VRay under the linear evaluation protocol). One possible explanation was that, in order to get a better CG forensics performance, the diversity of considered manipulation operations (i.e., the difference between the 9-class and 6-class versions), would be more important than the subdivision of manipulations with different parameters/settings (i.e.,

the difference between the 23-class and 9-class versions). The observations gained from this set of experiments would be useful for our future work on further experimental studies in this direction, as well as on the design of more appropriate pretext tasks for image forensics applications. For instance, and as mentioned above, it appears that increasing the diversity and types of manipulation operations would be more favorable than subdividing one manipulation into several with different intensities. In light of this observation, our proposed self-supervised pretext task could be further improved by seeking and including more kinds of diverse manipulation operations, or even by attempting to automatically derive the "optimal" set of manipulations (e.g., with inspiration gained from the method proposed in [55]).

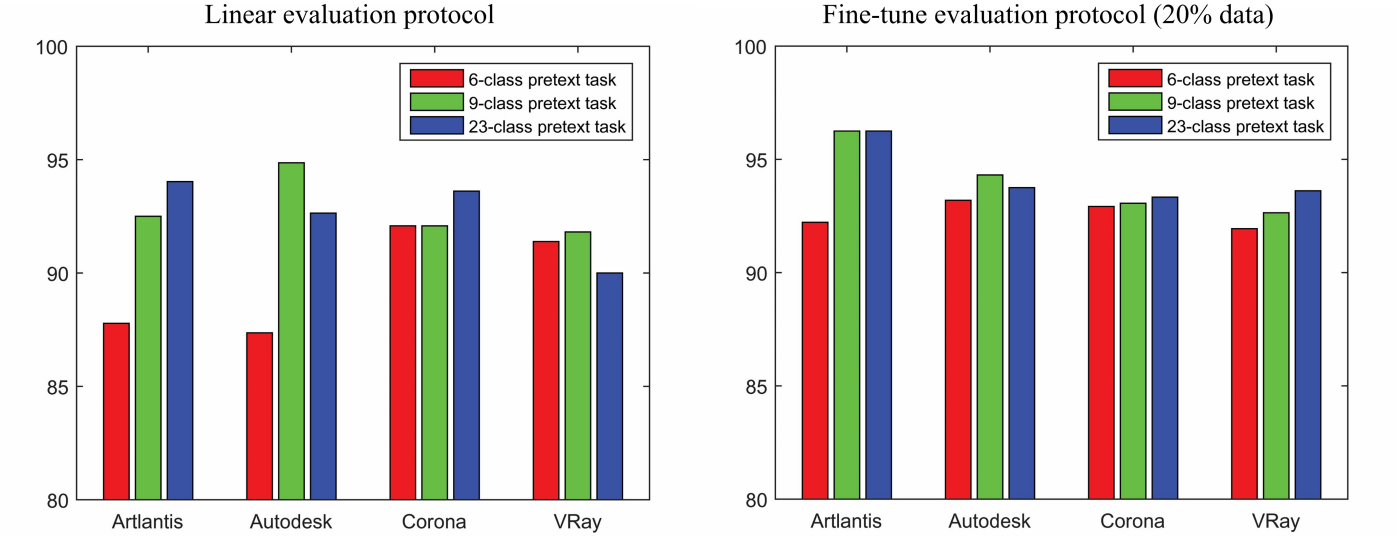

Figure 5. Comparisons of CG forensics performances of different versions of self-supervised pretext tasks, respectively, with 6 classes, 9 classes and 23 classes (*cf.* the main text for details). The comparisons were carried out on the four CG forensics datasets of Artlantis, Autodesk, Corona and VRay, under the linear evaluation protocol (left panel) and the fine-tune evaluation protocol (right panel). The vertical axes in the two panels represent the classification accuracy (in percentage) on testing set. The results in the two panels were not directly comparable, because the amount of training samples was different (whole training set for the left panel and 20% training data for the right panel).

In addition, one may be interested in the relative importance of manipulations on the performance of CG forensics. In practice, we found that the relevant study was rather complicated, because the performance depended on a number of factors, e.g., the considered dataset. For instance, in a set of preliminary experiments, we compared the performance of jointly modifying color saturation and color hue within one manipulation (as in our pretext task, *cf.* Table 1 row of "Color jitter") and that of separately applying modification on color saturation and on color hue as two manipulations. The obtained forensic accuracies under the fine-tuning evaluation protocol (with 20% training data) on the four datasets of Artlantis, Autodesk, Corona, and VRay were, respectively, 96.25% vs. 94.44%, 94.31% vs. 93.47%, 93.06% vs. 94.58%, and 92.64% vs. 93.47%, with the former number for joint modification and the latter one for separate modification. We can observe that there was no clear trend in these experimental results (although joint modification, as used in our 9-class pretext task, had slightly higher overall accuracy averaged on four datasets): the joint modification performed better on Artlantis and Autodesk, while the separate modification performed better on Corona and VRay. In another set of experiments, we compared two variants of 6-class pretext task, i.e., the first one with the removal of histogram equalization, pixel value quantization and Gamma correction from the 9-class pretext task (the same as in the previous paragraph) and the second one with the removal of Gaussian blurring, noise addition and sharpness enhancement. The obtained test accuracies under the fine-

tuning evaluation protocol (with 20% training data) on the four datasets were, respectively, 92.22% vs. 93.47%, 93.19% vs. 94.17%, 92.92% vs. 91.81%, and 91.94% vs. 91.94% (the former for the first variant, while the latter was for the second variant; a minor side remark here is that both 6-class variants had lower accuracy than the 9-class pretext task on all four datasets). We can observe that the second variant had better performance on Artlantis and Autodesk, while the first variant worked better on Corona, and that the performance was equal on VRay. It still remained difficult to assess the relative importance of the manipulations of histogram equalization, pixel value quantization and Gamma correction (i.e., straightforward modification of pixel values) and the manipulations of Gaussian blurring, noise addition and sharpness enhancement (i.e., rather frequency-oriented operations). Although facing a complicated research problem, we plan to make efforts in future studies to better understand these experimental observations, as well as the impact and relative importance of various manipulation operations. This would also allow us to gain deeper understanding of the real difference between CG images and NIs.

4.4.2. Self-Supervised Training on Whole ImageNet Dataset

The experimental results presented until now were based on a self-supervised pretext task training on 10% ImageNet data (about 128 K images). Such a pretext task training took about 14 h by using two advanced GPUs and, in most cases, it allowed us to reach decent CG forensics performances. It would be possible and beneficial to use more data for the pretext task training, which, however, would increase the training time with higher requirements of computing resources. A pretext task training with the whole ImageNet dataset took nearly 6 days on the two GPUs and would last about two weeks if only one ordinary GPU was available. Considering the repetitive procedure of the design and test of pretext task (several iterations were necessary), it would be quite expensive and time-consuming to use all ImageNet for pre-training, especially for researchers and practitioners having rather limited computing resources. Luckily, we were able to carry out one training of our 9-class classification pretext task on the whole ImageNet dataset and evaluated the pre-trained network for the main task of CG forensics. Table 6 presents the obtained experimental results, with comparisons to existing methods, under the fine-tuning evaluation protocol for the multi-class CG forensic problem. Here, for comparisons, we downloaded and used the pre-trained (on whole ImageNet) models of RotNet, Jigsaw, BarlowTwins, SimCLR, MoCo-V2 and ImageNetClassi from the VISSL repository [51]. From Table 6 we can see that our method still performed better than existing self-supervised schemes. It can also be observed that the results in Table 6 were better than those in Table 5 and that the pretext task training with more data (though being more time-consuming, more expensive and less ecological) was beneficial in terms of the performance on the main task of CG forensics. For example, the performance gap between our self-supervised method and the supervised ENet was very small under the fine-tuning on whole training set, and, as shown in the last column of Table 6, several self-supervised methods (including ours) achieved better performance than the supervised ENet under the fine-tuning on 20% data. In practice, for the design and implementation of a self-supervised learning method, it is recommended to find a good trade-off between different factors, like main task performance, computational time, affordability of computing hardware and energy consumption.

Table 6. Test accuracy of different methods under the fine-tuning evaluation protocol for the 5-class classification problem of CG forensics (the 5 classes being NI, CG of Artlantis, CG of Autodesk, CG of Corona, and CG of VRay). The presented results are classification accuracies on the testing set of a dataset properly constructed from the four datasets of Artlantis, Autodesk, Corona and VRay (*cf.* the last paragraph of Section 4.3 for details). The pretext task training of all self-supervised methods and the training of the ImageNetClassi method were carried out on the whole ImageNet dataset.

Method	Fine-Tune on Whole Train Set	Fine-Tune on 20% Train Set
RotNet [38]	79.83%	62.89%
Jigsaw [41]	82.17%	66.56%
BarlowTwins [46]	85.50%	73.78%
SimCLR [42]	85.78%	73.44%
MoCo-V2 [44]	86.33%	73.89%
ImageNetClassi	82.22%	66.00%
ENet adapted (supervised)	90.61%	71.06%
Ours	89.22%	79.56%

4.5. Discussion

The experimental results presented in this section showed that existing self-supervised methods, mostly designed for image content understanding, could, in some cases, reach rather satisfying performances for the main task of discriminating between CG images and NIs, but better results were achieved by our proposed method, especially in challenging scenarios with data scarcity and/or a more difficult forensic problem (e.g., when dealing with a multi-class CG forensic problem with few data available for fine-tuning). This was probably due to the better appropriateness of the self-supervised pretext task in our method, having the objective, as discussed in Section 3, to capture different kinds of deviations from NIs. This design was reasonable because, intuitively, CG images are a group of synthetic images that deviate, though sometimes subtly, from NIs. As shown in Section 3, the manipulation operations considered in our pretext task could introduce different types of visual distortions in the spatial domain (Figure 2), as well as different kinds of changes in the image pixel value histogram (Figure 3) and the joint distribution of neighboring pixels (Figure 4). Learned representations that were sensitive to these various modifications would be useful for detecting the subtle deviation of CG images from NIs. Our method had lower CG forensics accuracy than the state-of-the-art fully supervised method ENet in the majority of tested scenarios. This was rather understandable, and in line with the observation that early self-supervised methods for computer vision tasks have clearly lower performance than their supervised counterparts. Nevertheless, our method was able to achieve comparable, or even better. performance compared with ENet in the following three scenarios: the fine-tuning (on 20% data) binary classification scenario with the same number of training epochs (as mentioned towards the end of the first paragraph of Section 4.3), the challenging multi-class CG forensic problem with fine-tuning on 20% data (as shown in the last column of Table 5), and the same multi-class fine-tuning setting but with our method pre-trained on more data (as shown in the last column of Table 6, our method also clearly performed better). These results implied that our self-supervised method would have a good chance to outperform supervised methods when facing a difficult forensic problem and/or when few labeled samples were available. In addition, extensive self-supervised training is also beneficial if we can afford the time and computational costs. In our opinion, the experimental results of our method are encouraging, especially considering that this was. to our knowledge, the first attempt in the literature on self-supervised learning for CG forensics, and we think that our method would be useful in challenging practical cases; for instance, in which we have very few labeled data and/or we have a good number of possible sources of CG images (from different rendering engines) or even NIs (from different cameras) to identify. We believe that our self-supervised method, or its improved version, has the potential to reach high flexibility and good practical applicability in real-world forensic scenarios. Furthermore, it appears

that a good diversity of deviations considered in our pretext task would be beneficial for achieving better CG forensic performances. We plan to carry out studies to achieve better understanding on this point and design more suitable pretext tasks for the classification of CG images and NIs. It would also be interesting to explore the combination of our pretext task-based learning method with recent and popular ideas of general representation learning, such as restorative and adversarial learning [56], disentangled representation learning [57] and masked modeling [58,59].

5. Conclusions and Future Work

In this paper, we presented, to our knowledge, the first study in the literature on the utility of self-supervised learning for the discrimination of CG images and NIs. We proposed a new self-supervised learning method with a forensics-oriented pretext task in which the objective was to correctly classify authentic images and their modified versions obtained after applying different kinds of manipulations. The self-supervised pre-training was carried out without leveraging any human manual labeling effort and without using any CG image. Experimental results under different evaluation scenarios demonstrated that our method coped better with the CG forensics problem than existing self-supervised methods. In a scenario with data scarcity and a difficult forensic problem, our self-supervised method could achieve comparable, or even better, performance compared with a specially designed supervised method. We hope that this first attempt on self-supervised CG forensics could encourage more research efforts in this interesting and promising working direction. One limitation of our method was that there was still a performance gap when compared with a state-of-the-art fully supervised method, especially under situations with a relatively easy binary classification problem and with sufficient training data. This, however, remains understandable, and is similar to the existence of performance gap between early self-supervised methods and their supervised counterparts for computer vision applications. In order to enhance the practical utility of our method, we plan to conduct studies on the improvement of the learning capacity of the self-supervised framework, e.g., by combining forensics-oriented pretext task learning with the idea of metric learning related to the perceptual quality of images. Our proposed method makes the assumption (though quite reasonable) that only NIs are available for the self-supervised learning. If CG images are also available, we might make use of these images to further improve the model obtained after self-supervised learning with an additional training step, in an attempt to make the model more adapted to the data and problems that may be encountered in real-world applications. In this case, we could consider also applying manipulations on CG images, either as an effective data augmentation technique (i.e., processed CG images as augmented and additional training data), or in a contrastive learning framework (e.g., two augmented versions of a CG image should have consistent features and prediction results). In a broad sense, we would like to carry out studies on the design of better self-supervised methods for CG forensics, with a more suitable pretext task or even with a brand-new learning scheme. For example, it might be interesting to derive pretext tasks and self-supervised learning schemes related to the statistical modeling of natural images or to the perception of visual appearance of authentic and degraded images. The combination of these different self-supervised learning schemes, e.g., within a multi-task learning framework, would allow us to achieve improved performance for the discrimination of CG images and NIs. Finally, we would like to explore the utility of the self-supervised learning mechanism for other related applications, including the detection of image forgeries (e.g., splicing, copy-move and inpainting) and the assessment of visual or aesthetic quality of digital images.

Funding: This work was partially funded by the French National Research Agency grant number ANR-15-IDEX-02.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The author would like to thank NVIDIA for a GPU gift.

Conflicts of Interest: The author declares no conflict of interest. The funder had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Farid, H. Digital Image Forensics, 2012. Tutorial and Course Notes. Available online: https://farid.berkeley.edu/downloads/tutorials/digitalimageforensics.pdf (accessed on 27 December 2022).
2. Piva, A. An overview on image forensics. *Isrn Signal Process.* **2013**, *2013*, 496701. [CrossRef]
3. Verdoliva, L. Media forensics and deepfakes: An overview. *IEEE J. Sel. Top. Signal Process.* **2020**, *14*, 910–932. [CrossRef]
4. Castillo Camacho, I.; Wang, K. A comprehensive review of deep-learning-based methods for image forensics. *J. Imaging* **2021**, *7*, 69. [CrossRef] [PubMed]
5. Ng, T.T.; Chang, S.F. Discrimination of Computer Synthesized or Recaptured Images from Real Images. In *Digital Image Forensics*; Sencar, H.T., Memon, N., Eds.; Springer: New York, NY, USA, 2013; pp. 275–309.
6. Quan, W.; Wang, K.; Yan, D.M.; Zhang, X. Distinguishing between natural and computer-generated images using convolutional neural networks. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 2772–2787. [CrossRef]
7. Yang, P.; Baracchi, D.; Ni, R.; Zhao, Y.; Argenti, F.; Piva, A. A survey of deep learning-based source image forensics. *J. Imaging* **2020**, *6*, 9. [CrossRef]
8. Jing, L.; Tian, Y. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 4037–4058. [CrossRef]
9. Chaosgroup Gallery. Available online: https://www.chaosgroup.com/gallery (accessed on 27 December 2022).
10. Learn V-Ray Gallery. Available online: https://www.learnvray.com/fotogallery/ (accessed on 27 December 2022).
11. Corona Renderer Gallery. Available online: https://corona-renderer.com/gallery (accessed on 27 December 2022).
12. Shullani, D.; Fontani, M.; Iuliani, M.; Shaya, O.A.; Piva, A. VISION: A video and image dataset for source identification. *Eurasip J. Inf. Secur.* **2017**, *2017*, 15. [CrossRef]
13. Dang-Nguyen, D.T.; Pasquini, C.; Conotter, V.; Boato, G. RAISE: A raw images dataset for digital image forensics. In Proceedings of the ACM Multimedia Systems Conference, Portland, OR, USA, 18–20 March 2015; pp. 219–224.
14. Ng, T.T.; Chang, S.F.; Hsu, J.; Xie, L.; Tsui, M.P. Physics-motivated features for distinguishing photographic images and computer graphics. In Proceedings of the ACM International Conference on Multimedia, Singapore, Singapore, 6–11 November 2005; pp. 239–248.
15. Pan, F.; Chen, J.; Huang, J. Discriminating between photorealistic computer graphics and natural images using fractal geometry. *Sci. China Ser. Inf. Sci.* **2009**, *52*, 329–337. [CrossRef]
16. Zhang, R.; Wang, R.D.; Ng, T.T. 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, Shanghai, China, 31 October–3 November 2012; pp. 292–305.
17. Peng, F.; Zhou, D.L. Discriminating natural images and computer generated graphics based on the impact of CFA interpolation on the correlation of PRNU. *Digit. Investig.* **2014**, *11*, 111–119. [CrossRef]
18. Sankar, G.; Zhao, V.; Yang, Y.H. Feature based classification of computer graphics and real images. In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Taipei, Taiwan, 19–24 April 2009; pp. 1513–1516.
19. Lyu, S.; Farid, H. How realistic is photorealistic? *IEEE Trans. Signal Process.* **2005**, *53*, 845–850. [CrossRef]
20. Wang, J.; Li, T.; Shi, Y.Q.; Lian, S.; Ye, J. Forensics feature analysis in quaternion wavelet domain for distinguishing photographic images and computer graphics. *Multimed. Tools Appl.* **2017**, *76*, 23721–23737. [CrossRef]
21. Chen, W.; Shi, Y.Q.; Xuan, G. Identifying computer graphics using HSV color model and statistical moments of characteristic functions. In Proceedings of the IEEE International Conference on Multimedia & Expo, Beijing, China, 2–5 July 2007; pp. 1123–1126.
22. Özparlak, L.; Avcibas, I. Differentiating between images using wavelet-based transforms: A comparative study. *IEEE Trans. Inf. Forensics Secur.* **2011**, *6*, 1418–1431. [CrossRef]
23. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
24. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016. Available online: http://www.deeplearningbook.org (accessed on 27 December 2022).
25. Rahmouni, N.; Nozick, V.; Yamagishi, J.; Echizen, I. Distinguishing computer graphics from natural images using convolution neural networks. In Proceedings of the IEEE International Workshop on Information Forensics and Security, Rennes, France, 4–7 December 2017; pp. 1–6.
26. Yao, Y.; Hu, W.; Zhang, W.; Wu, T.; Shi, Y.Q. Distinguishing computer-generated graphics from natural images based on sensor pattern noise and deep learning. *Sensors* **2018**, *18*, 1296. [CrossRef]

27. Quan, W.; Wang, K.; Yan, D.M.; Zhang, X.; Pellerin, D. Learn with diversity and from harder samples: Improving the generalization of CNN-based detection of computer-generated images. *Forensic Sci. Int. Digit. Investig.* **2020**, *35*, 301023. [CrossRef]
28. He, P.; Li, H.; Wang, H.; Zhang, R. Detection of computer graphics using attention-based dual-branch convolutional neural network from fused color components. *Sensors* **2020**, *20*, 4743. [CrossRef] [PubMed]
29. Zhang, R.; Quan, W.; Fan, L.; Hu, L.; Yan, D.M. Distinguishing computer-generated images from natural images using channel and pixel correlation. *J. Comput. Sci. Technol.* **2020**, *35*, 592–602. [CrossRef]
30. Bai, W.; Zhang, Z.; Li, B.; Wang, P.; Li, Y.; Zhang, C.; Hu, W. Robust texture-aware computer-generated image forensic: Benchmark and algorithm. *IEEE Trans. Image Process.* **2021**, *30*, 8439–8453. [CrossRef] [PubMed]
31. Yao, Y.; Zhang, Z.; Ni, X.; Shen, Z.; Chen, L.; Xu, D. CGNet: Detecting computer-generated images based on transfer learning with attention module. *Signal Process. Image Commun.* **2022**, *105*, 116692. [CrossRef]
32. Nguyen, H.H.; Yamagishi, J.; Echizen, I. Capsule-forensics: Using Capsule networks to detect forged images and videos. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Brighton, UK, 12–17 May 2019; pp. 2307–2311.
33. He, P.; Jiang, X.; Sun, T.; Li, H. Computer graphics identification combining convolutional and recurrent neural networks. *IEEE Signal Process. Lett.* **2018**, *25*, 1369–1373. [CrossRef]
34. Bhalang Tarianga, D.; Senguptab, P.; Roy, A.; Subhra Chakraborty, R.; Naskar, R. Classification of computer generated and natural images based on efficient deep convolutional recurrent attention model. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Long Beach, CA, USA, 15–20 June 2019; pp. 146–152.
35. Liu, S.; Mallol-Ragolta, A.; Parada-Cabaleiro, E.; Qian, K.; Jing, X.; Kathan, A.; Hu, B.; Schuller, B.W. Audio self-supervised learning: A survey. *Patterns* **2022**, *3*, 100616. [CrossRef]
36. Krishnan, R.; Rajpurkar, P.; Topol, E.J. Self-supervised learning in medicine and healthcare. *Nat. Biomed. Eng.* **2022**, *6*, 1346–1352. [CrossRef] [PubMed]
37. Liu, Y.; Jin, M.; Pan, S.; Zhou, C.; Zheng, Y.; Xia, F.; Yu, P. Graph self-supervised learning: A survey. *IEEE Trans. Knowl. Data Eng.* **2022**, 1–20. [CrossRef]
38. Gidaris, S.; Singh, P.; Komodakis, N. Unsupervised representation learning by predicting image rotations. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018; pp. 1–16.
39. Zhang, R.; Isola, P.; Efros, A.A. Colorful image colorization. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 649–666.
40. Doersch, C.; Gupta, A.; Efros, A.A. Unsupervised visual representation learning by context prediction. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1422–1430.
41. Noroozi, M.; Favaro, P. Unsupervised learning of visual representations by solving jigsaw puzzles. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 69–84.
42. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G.E. A simple framework for contrastive learning of visual representations. In Proceedings of the International Conference on Machine Learning, Virtual Event, 12–18 July 2020; pp. 1597–1607.
43. He, K.; Fan, H.; Wu, Y.; Xie, S.; Girshick, R. Momentum contrast for unsupervised visual representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 9729–9738.
44. Chen, X.; Fan, H.; Girshick, R.; He, K. Improved baselines with momentum contrastive learning. *CoRR* **2020**, 1–3. [CrossRef]
45. Grill, J.B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P.H.; Buchatskaya, E.; Doersch, C.; Pires, B.A.; Guo, Z.D.; Azar, M.G.; et al. Bootstrap your own latent: A new approach to self-supervised learning. In Proceedings of the Advances in Neural Information Processing Systems, Virtual Event, 6–12 December 2020; pp. 21271–21284.
46. Zbontar, J.; Jing, L.; Misra, I.; LeCun, Y.; Deny, S. Barlow Twins: Self-supervised learning via redundancy reduction. In Proceedings of the International Conference on Machine Learning, Virtual Event, 18–24 July 2021; pp. 12310–12320.
47. Bayar, B.; Stamm, M.C. Constrained convolutional neural networks: A new approach towards general purpose image manipulation detection. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 2691–2706. [CrossRef]
48. Castillo Camacho, I.; Wang, K. Convolutional neural network initialization approaches for image manipulation detection. *Digit. Signal Process.* **2022**, *122*, 103376. [CrossRef]
49. Deng, J.; Dong, W.; Socher, R.; Li, L.; Li, K.; Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
50. Hyvärinen, A.; Hurri, J.; Hoyer, P.O. *Natural Image Statistics—A Probabilistic Approach to Early Computational Vision*; Springer: London, UK, 2009.
51. Goyal, P.; Duval, Q.; Reizenstein, J.; Leavitt, M.; Xu, M.; Lefaudeux, B.; Singh, M.; Reis, V.; Caron, M.; Bojanowski, P.; et al. VISSL (Computer VIsion Library for State-of-the-Art Self-Supervised Learning). 2021. Available online: https://github.com/facebookresearch/vissl (accessed on 27 December 2022).
52. Artlantis Gallery. Available online: https://artlantis.com/en/gallery/ (accessed on 27 December 2022).
53. Autodesk A360 Rendering Gallery. Available online: https://gallery.autodesk.com/a360rendering/ (accessed on 27 December 2022).
54. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
55. Xie, Q.; Dai, Z.; Hovy, E.; Luong, M.T.; Le, Q.V. Unsupervised data augmentation for consistency training. In Proceedings of the Advances in Neural Information Processing Systems, Virtual Event, 6–12 December 2020; pp. 6256–6268.

56. Haghighi, F.; Taher, M.R.H.; Gotway, M.B.; Liang, J. DiRA: Discriminative, restorative, and adversarial learning for self-supervised medical image analysis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 20824–20834.
57. Liu, X.; Sanchez, P.; Thermos, S.; O'Neil, A.Q.; Tsaftaris, S.A. Learning disentangled representations in the imaging domain. *Med. Image Anal.* **2022**, *80*, 102516. [CrossRef]
58. Tong, Z.; Song, Y.; Wang, J.; Wang, L. VideoMAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In Proceedings of the Advances in Neural Information Processing Systems, New Orleans, LA, USA, 28 November–9 December 2022; pp. 1–15.
59. Wei, C.; Fan, H.; Xie, S.; Wu, C.Y.; Yuille, A.; Feichtenhofer, C. Masked feature prediction for self-supervised visual pre-training. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 14668–14678.