Noise Doesn’t Lie: Towards Universal Detection of Deep Inpainting

Ang Li\textsuperscript{1}, Qiuhong Ke\textsuperscript{1}, Xingjun Ma\textsuperscript{2*}, Haiqin Weng\textsuperscript{3}, Zhiyuan Zong\textsuperscript{3}, Feng Xue\textsuperscript{3} and Rui Zhang\textsuperscript{4}

\textsuperscript{1}The University of Melbourne
\textsuperscript{2}Deakin University
\textsuperscript{3}Ant Group
\textsuperscript{4}Tsinghua University

angl4@student.unimelb.edu.au, qiuhong.ke@unimelb.edu.au, daniel.ma@deakin.edu.au, haiqin.wenghaiqin@antfin.com, david.zzy@antgroup.com, gkn1fexxx@gmail.com, rayteam@yeah.net

Abstract

Deep image inpainting aims to restore damaged or missing regions in an image with realistic contents. While having a wide range of applications such as object removal and image recovery, deep inpainting techniques also have the risk of being manipulated for image forgery. A promising countermeasure against such forgeries is deep inpainting detection, which aims to locate the inpainted regions in an image. In this paper, we make the first attempt towards \textit{universal} detection of deep inpainting, where the detection network can generalize well when detecting \textit{different} deep inpainting methods. To this end, we first propose a novel data generation approach to generate a \textit{universal} training dataset, which imitates the noise discrepancies exist in real versus inpainted image contents to train universal detectors. We then design a Noise-Image Cross-fusion Network (NIX-Net) to effectively exploit the discriminative information contained in both the images and their noise patterns. We empirically show, on multiple benchmark datasets, that our approach outperforms existing detection methods by a large margin and generalize well to unseen deep inpainting techniques. Our universal training dataset can also significantly boost the generalizability of existing detection methods.

1 Introduction

Image Inpainting is the process of restoring damaged or missing regions of a given image based on the information of the undamaged regions. It has a wide range of real-world applications such as the restoration of damaged images and the removal of unwanted objects. So far, plenty of inpainting approaches have been proposed, among which generative adversarial networks (GANs) \cite{Goodfellow2014} based deep inpainting techniques \cite{Pathak2016, Iizuka2017, Yu2018, Li2019a, Yu2019, Li2020} have been demonstrated to be the most effective ones. One distinguishing advantage of deep inpainting models is the ability to adaptively predict semantic structures and produce super realistic and fine-detailed textures.

However, like a two-edged sword cuts both ways, deep inpainting techniques come along with the risk of being manipulated for image forgery. Due to the super realistic inpainting effects, these techniques can be easily applied to replace the critical objects in an image with fake contents, and the tampered image may appear as photo-realistic as real images. Figure 1 (b) shows one such example crafted from the real image in Figure 1 (a) by a recent deep inpainting method \cite{Yu2019}. Inpainted images can potentially be used to create fake news, spread rumors on the internet or even fabricate false evidences. It is thus imperative to develop detection algorithms to identify whether and more importantly where an image has been modified by deep inpainting. Specifically, the goal of deep inpainting detection is to locate the exact inpainted regions in an image, as shown in Figure 1 (d).

While Li et al. \cite{Li2019} recently proposed the first method for deep inpainting detection, its effectiveness is restricted to the inpainting technique the detector was trained on and does not generalize well to other inpainting techniques. However, in real-world scenarios, the exact techniques used to inpaint the images are often unknown. In this paper, we aim to address this generalization limitation and introduce a \textit{universal} detector that works well even on unseen deep inpainting techniques.

Our approach is motivated by one important yet so far overlooked common characteristic of all deep inpainting methods: the patterns of the noise exists in real and synthesized contents are different. It has been shown that conventional image acquisition devices (e.g. camera sensors) leave dis-
In summary, our main contributions are:

- We propose a novel framework for universal deep inpainting detection, which consists of 1) a new method of generating universal training data, and 2) a two-stream multi-scale Noise-Image Cross-fusion detection Network (NIX-Net).
- We empirically show, on multiple benchmark datasets, that our proposed approach can consistently outperform existing detection methods, especially when applied to detect unseen deep inpainting techniques.
- Our universal training dataset can also improve the generalizability of existing detection methods, making it an indispensable part of future detection methods.

2 Related Work

2.1 Deep Image Inpainting

Different from conventional image inpainting approaches, deep learning based image inpainting (or deep inpainting for short) trains inpainting networks on large-scale datasets and can generate more visually plausible details or fill large missing regions with new contents that never exist in the input image. By far, the generative adversarial networks (GAN) [Goodfellow et al., 2014] based inpainting methods are arguably the most powerful methods for deep image inpainting. These methods all employ a GAN-based training approach with two sub-networks: an inpainting network and a discriminative network. The former learns image semantics and fills the missing regions with predicted contents, whereas the latter distinguishes whether the image is real or inpainted.

Phatak et al. [2016] proposed the Context-Encoder (CE) for single image inpainting, which is known as the first GAN-based image inpainting technique. This technique was later improved by [Iizuka et al., 2017] using dilated convolution and global-local adversarial training. Yu et al. [2018] proposed a two-stage inpainting network with a coarse-to-fine learning strategy. A gated convolution network along with a learnable dynamic feature selection mechanism (for each channel and at each spatial location) was proposed in [Yu et al., 2019] for image inpainting. Li et al. [2020a] devised the Recurrent Feature Reasoning network which recurrently enriches information for the hole region. Despite the diversity of existing deep inpainting methods, they all share a common characteristic: the noise patterns in generated contents are different from those in the real image contents. While this characteristic has been observed in understanding the artificial fingerprints of GANs [Marra et al., 2019], it has not been exploited for deep inpainting detection. In this paper, we will leverage such a universal characteristic of generated contents to build universal deep inpainting detectors.

2.2 Inpainting Forensics

Deep inpainting detection falls into the general scope of image forensics, but quite different from the conventional image manipulation detection or deepfake detection. Conventional image manipulation detection deals with traditional image forgery operations such as splicing [Huh et al., 2018] and copy-move [Wu et al., 2018]. Deepfake (or deep face swapping) is the other type of deep learning forgery techniques that swaps one person’s face in a video to that of a different person, which often requires heavy post-processing including color transfer and boundary blending [Li et al., 2020b]. Different from conventional image manipulation or deepfake, deep inpainting takes one image and a mask as inputs and generates new content for the mask region, based on information of the non-mask regions within the same image. In this paper, we focus on deep image inpainting detection and the generalizability of the detector to unseen deep inpainting techniques (not to conventional image manipulation or deepfake).

Most of existing inpainting forensic methods are developed to detect traditional image inpainting techniques. For example, the detection of traditional diffusion-based inpainting based on local variance of image Laplacian [Li et al., 2017], and the detection of traditional patch-based inpainting via patch similarities computed by zero-connectivity length [Wu et al., 2008], two-stage suspicious region search [Chang et al., 2013] or CNN-based encoder-decoder detection networks [Zhu et al., 2018]. These methods are generally less effective on deep inpainting techniques that can synthesize extremely photo-realistic contents or new objects that never exist in the original image. Deep inpainting detection is a
we will introduce our proposed universal training dataset generation. The detection model to unseen deep inpainting methods. Next, we propose to generate a training dataset by simulating “inpainted” images from autoencoder reconstructions, instead of using any existing deep inpainting methods. The complete generation procedure is illustrated in Figure 3.

For a set of real images \( I \) and inpainting methods, the generation process generates a set of simulated images \( \{I_1, I_2, ..., I_n\} \) using a pre-trained autoencoder with a set of random binary masks \( M \) and \( \{M_1, M_2, ..., M_n\} \). Specifically, given a real image \( I_i \), we first obtain its reconstructed version \( G(I_i) \) from an autoencoder \( G \). We train the autoencoder following a typical GAN approach using the autoencoder as the GAN generator and an additional classification network as the GAN discriminator. The overall structure is illustrated in Figure 4. The autoencoder is trained to have small reconstruction error, and at the same time, the reconstructed images should be as realistic as the real images according to the discriminator. The overall training loss of this autoencoder is:

\[
L = \sum_{i=1}^{n} \log(D(I_i)) + \log(1 - D(G(I_i))) + \lambda \|G(I_i) - I_i\|_2
\]

where, \( L_{adv} \) and \( L_{rec} \) are the adversarial and reconstruction loss respectively, and \( \lambda = 0.1 \) is a trade-off parameter.

After training, the autoencoder \( G \) is applied to reconstruct each real image in \( I \). With the reconstructed images, our next step is to simulate the inpainting process. Specifically, we simulate an “inpainted” image \( X_i \) by combining \( I_i \) and its reconstruction \( G(I_i) \) according to a random mask \( M_i \):

\[
X_i = M_i \odot I_i + (1 - M_i) \odot G(I_i)
\]

where \( \odot \) is the element-wise multiplication and \( M_i \) is a binary mask with 0 elements indicating the inpainting region (white region) and 1 elements indicating the non-inpainting region (black region). In \( X_i \), the “inpainted” region carries over the noise patterns of the synthesized contents from the autoencoder, while the rest of the regions preserve the noise patterns from the real contents. Following this procedure, we can obtain a set of images with synthesized regions to create the universal training dataset: \( UT = \{X, M\} \), with \( X \) the simulated inpainted images and \( M \) being the inpainting masks.

Our universal dataset \( UT \) distinguishes itself from existing inpainting-method-aware datasets as a general formulation of real versus generated contents. However, having this dataset is not enough to train accurate deep inpainting detectors. It requires an effective learning framework to exploit the discriminative information contained in the dataset, especially the noise patterns. Next we will introduce our proposed detection network that serves this purpose.

3 Problem Formulation and Universal Training Dataset Generation

3.1 Problem Formulation

Given an image \( X_i \) inpainted by a certain deep inpainting method on regions defined by a binary mask \( M_i \), deep inpainting detection aims to locate the inpainted regions \( M_i \). A detection network can be trained to take the inpainted image \( X_i \) as input and output the predicted mask \( \hat{M}_i \).

To train the detector, a straightforward approach is to use inpainted images generated by a deep inpainting method as the training data [Li and Huang, 2019]. However, this inpainting-method-aware approach often generalizes poorly when applied to detected inpainted images generated by unseen deep inpainting methods. An empirically analysis can be found in Section 5.1. In contrast, we propose to generate a universal training dataset to capture the common characteristics shared by different deep inpainting methods, and train the detection model on this universal dataset. Such an inpainting-method-agnostic approach can improve the generalization of the detection model to unseen deep inpainting methods. Next, we will introduce our proposed universal training dataset generation method inspired by the distinctive noise patterns exist in real versus generated contents.

3.2 Universal Training Dataset Generation

A universal training dataset should consider the common characteristics of different deep inpainting methods, rather than relying on the specific artifacts of one particular deep inpainting method. Motivated by our observation in Figure 2, here we propose to create the universal training dataset by simulating “inpainted” images from autoencoder reconstructions, instead of using any existing deep inpainting methods. An empirically analysis can be found in Section 5.1. In contrast, we propose to generate a universal training dataset: \( UT = \{X, M\} \), with \( X \) being simulated inpainted images and \( M \) being the inpainting masks.

The complete generation procedure is illustrated in Figure 3.

To train the detector, a straightforward approach is to use real images according to the discriminator. The overall training loss of this autoencoder is:

\[
L = \sum_{i=1}^{n} \log(D(I_i)) + \log(1 - D(G(I_i))) + \lambda \|G(I_i) - I_i\|_2
\]

where, \( L_{adv} \) and \( L_{rec} \) are the adversarial and reconstruction loss respectively, and \( \lambda = 0.1 \) is a trade-off parameter.

After training, the autoencoder \( G \) is applied to reconstruct each real image in \( I \). With the reconstructed images, our next step is to simulate the inpainting process. Specifically, we simulate an “inpainted” image \( X_i \) by combining \( I_i \) and its reconstruction \( G(I_i) \) according to a random mask \( M_i \):

\[
X_i = M_i \odot I_i + (1 - M_i) \odot G(I_i)
\]

where \( \odot \) is the element-wise multiplication and \( M_i \) is a binary mask with 0 elements indicating the inpainting region (white region) and 1 elements indicating the non-inpainting region (black region). In \( X_i \), the “inpainted” region carries over the noise patterns of the synthesized contents from the autoencoder, while the rest of the regions preserve the noise patterns from the real contents. Following this procedure, we can obtain a set of images with synthesized regions to create the universal training dataset: \( UT = \{X, M\} \), with \( X \) being simulated inpainted images and \( M \) being the inpainting masks.

Our universal dataset \( UT \) distinguishes itself from existing inpainting-method-aware datasets as a general formulation of real versus generated contents. However, having this dataset is not enough to train accurate deep inpainting detectors. It requires an effective learning framework to exploit the discriminative information contained in the dataset, especially the noise patterns. Next we will introduce our proposed detection network that serves this purpose.
4 Noise-Image Cross-fusion Network (NIX-Net)

As shown in Figure 2 (c), the noise residual patterns between inpainted and non-inpainted regions in an inpainted image are distinct. Unlike previous works that ignore this important cue for mask detection, we propose NIX-Net which leverages both the inpainted image $X_i$ and its noise residual $R_i$ to enhance detection performance. As shown in Figure 5, the proposed NIX-Net consists of three components: 1) feature extraction, 2) multi-scale cross fusion, and 3) mask detection.

**Feature extraction.** Given an inpainted image $X_i$, we define its noise residual as following:

$$R_i = X_i - d(X_i),$$  \hspace{1cm} (3)

Inspired by recent studies on image forensics using SRM features [Zhou et al., 2018], we choose the SRM filter as our denoising filter $d(\cdot)$. The feature extraction component includes two parallel feature extraction streams to learn multiscale feature maps of the input image and its noise residual, respectively. Each feature extraction module consists of three ResNet blocks [He et al., 2016b], resulting in three feature maps of $X_i$, namely, $\phi_{1i}(X_i)$, $\phi_{2i}(X_i)$ and $\phi_{3i}(X_i)$. The numbers of channels in the feature maps are 128, 256 and 512, respectively. Likewise, the three feature maps of $R_i$ are $\phi_{N1}(R_i)$, $\phi_{N2}(R_i)$ and $\phi_{N3}(R_i)$. Since the last convolution layer of each ResNet block has a stride of 2 to reduce the spatial scale [He et al., 2016a], the spatial scales of the feature maps are 1/2, 1/4 and 1/8 of the input spatial size.

**Multi-scale cross fusion.** The multi-scale cross fusion component aims to effectively incorporate noise features and image features in a multi-scale manner. Specifically, both the noise stream and the image stream are followed by a fusion module. The fusion module takes the three-scale feature maps as input, and outputs the crossly fused three-scale feature maps. Each fused feature map is the sum of the three transformed (upsampled, downsampled or unchanged) input feature maps. The outputs of fusion module 1 in the image stream are $\phi_{1i}^f(X_i)$, $\phi_{2i}^f(X_i)$ and $\phi_{3i}^f(X_i)$, while the outputs of fusion module 2 in the noise stream are $\phi_{N1}^f(R_i)$, $\phi_{N2}^f(R_i)$ and $\phi_{N3}^f(R_i)$. The purpose of this fusion module is to exchange the information across multi-scale feature representations, and produce richer feature representations with strengthened position sensitivity [Sun et al., 2019]. For each scale, we concatenate the feature maps of the image stream and the noise stream along the channel dimension, followed by a Conv block. The outputs are denoted as $\psi_1(X_i, R_i)$, $\psi_2(X_i, R_i)$ and $\psi_3(X_i, R_i)$. The Conv block consists of two 3 x 3 convolutions, each of which is followed by a batch normalization and a ReLU Layer. Finally, fusion module 3 further consolidates the connection between the image and noise features over different scales, and output three fused feature maps $\psi_1^f(X_i, R_i)$, $\psi_2^f(X_i, R_i)$ and $\psi_3^f(X_i, R_i)$.

**Mask detection.** The mask detection module first upsamples the two lower-resolution feature maps $\psi_2^f(X_i, R_i)$ and $\psi_3^f(X_i, R_i)$ using bilinear upsampling so that they have the same resolution as $\psi_1^f(X_i, R_i)$. The three feature maps are then concatenated along the channel dimension, followed by a Conv block and an upsampling layer which outputs a $w \times h \times 1$ feature map. The feature map is then fed into a Sigmoid layer for classification, rendering the possibility map $P_i$ with pixel-wise predictions. Finally, the detected mask $M_i$ can be obtained by binarizing $P_i$ according to a threshold value. In this paper, we set the threshold value as 0.5.

**Network training.** We train the entire network end-to-end using the focal loss [Lin et al., 2017] on the universal training
dataset \( UT \). Note that the network can also be trained on any other inpainting detection datasets including the inpainting-method-aware datasets. The use of the focal loss is to mitigate the effect of class imbalance (the inpainted regions are often small compared to the entire image). The focal loss \( \mathcal{L}_f \) is defined as following:

\[
\mathcal{L}_f = \sum_i (\tilde{M}_i (1 - \tilde{M}_i)^\gamma \log \tilde{M}_i - (1 - \tilde{M}_i) \tilde{M}_i^\gamma \log(1 - \tilde{M}_i))
\]

where \( \gamma \) is the focusing parameter and is set to 2.

5 Experiments

In this section, we first introduce the experimental settings, then evaluate the performance of our proposed approach via extensive experiments and ablation studies.

Inpainting methods and datasets. We use three different deep inpainting techniques including GL [Iizuka et al., 2017], CA [Yu et al., 2018] and GC [Yu et al., 2019] to generate inpainted images on two datasets Places2 [Zhou et al., 2017] and CelebA [Liu et al., 2015]. For each of the two datasets, we randomly select (without replacement) 50K, 10K and 10K images to create the training, validation and testing subsets respectively, following either our universal data generation or using one of the above three inpainting techniques (GL, CA and GC). We train the detection models on the training subset and test their performance on the test subset.

Mask generation. To simulate more diverse and complex real-world scenarios, we utilize the irregular mask setting in [Yu et al., 2019] with arbitrary shapes and random locations for both training and testing. Besides, object-shape masks are also adopted for visual comparison, as shown in Figure 6.

Baseline models. We consider two baseline models: 1) LDICN [Li and Huang, 2019], a fully convolutional network designed for deep inpainting detection; and 2) ManTra-Net [Wu et al., 2019], a state-of-the-art detection model for traditional image forgery such as splicing.

Performance metric. We use the Intersection over Union (IoU) as the performance metric, and report the mean IoU (mIoU) over the entire test subset of inpainted images.

Training setting. We train the networks using the Adam optimizer with initial learning rate \( 1 \times 10^{-4} \). An early stopping strategy is also adopted based on the mIoU on the validation dataset: the model with the highest validation mIoU is saved as the final model. All of our experiments were run with a Nvidia Tesla V100 GPU.

5.1 Quantitative Performance Evaluation

We have 3 detection networks (LDICN, ManTra-Net and our NIX-Net) and 2 types of training data including 1) 3 inpainting-method-aware training datasets created using 3 inpainting methods (GL, CA and GC), and 2) our universal (UT) training dataset. Here, we first train the detection networks on GL/CA/GC then test the performance on the test sets of all three datasets. Besides, we run our NIX-Net on a hybrid dataset that combines UT with one out of the 3 inpainting-method-aware datasets. We also train the detection networks on UT only to test the importance of universal training data to generalizability. Note that all these experiments are run separately for Places2 and CelebA. The performance of the 3 detection networks are summarized in Table 1.

Overall performance. As shown in Table 1, our NIX-Net outperforms existing methods by a large margin in all test scenarios, especially when transferred to detect unseen inpainting methods. When trained on the hybrid datasets, our NIX-Net achieved the best overall performance. Next, we will provide a detailed analysis of these results from two perspectives: 1) the effectiveness of different detection networks, and 2) the importance of universal training data.

Effectiveness of different detection networks. For LDICN and ManTra-Net, although decent results can be obtained on the known (used for generating training data) inpainting method, their performance drops drastically on unseen inpainting methods. Such a poor generalizability indicates that both models overfit to the artifacts of a particular inpainting method and fail to consider the common characteristics of different deep inpainting techniques. By contrast, our NIX-Net demonstrates consistently better generalizability, regardless of the inpainting method used for training. This is largely due to the sufficient (multi-scale and cross-fusion) exploitation of the noise information contained in real versus inpainted contents. This also indicates that noise patterns are indeed a reliable cue of detecting inpainted regions.

Importance of universal training data. Revisit Table 1, we find that, whenever the UT dataset is used in conjunction with one inpainting-method-aware dataset, the generalization performance of our NIX-Net can be significantly improved. Moreover, the UT dataset alone can lead to much better generalizability of existing methods LDICN and ManTra-Net. This result verifies, from the data perspective, the importance of noise modeling for universal deep inpainting detection. More importantly, such noise modeling like our proposed universal training dataset generation is much easier than gener-

<table>
<thead>
<tr>
<th>Model</th>
<th>Training Data</th>
<th>Test mIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Places2</td>
<td>CelebA</td>
</tr>
<tr>
<td></td>
<td>UT</td>
<td>GL CA GC</td>
</tr>
<tr>
<td></td>
<td>UT</td>
<td>GL CA GC</td>
</tr>
<tr>
<td>LDICN</td>
<td>✓</td>
<td>83.47</td>
</tr>
<tr>
<td>ManTra-Net</td>
<td>✓</td>
<td>88.76</td>
</tr>
<tr>
<td>NIX-Net</td>
<td>✓</td>
<td>91.82</td>
</tr>
<tr>
<td></td>
<td>✓</td>
<td>92.14</td>
</tr>
</tbody>
</table>

Table 1: Quantitative Comparison on Places2 and CelebA datasets.
5.3 Ablation Study

Here, we run a set of ablation studies to provide a complete understanding of the two key components of our NIX-Net network: two-stream (noise+image) feature learning and multi-scale cross fusion. Table 2 compares the full NIX-Net detection network with its five variants created by removing or keeping the noise/image stream or the three fusion modules. All these networks are trained on the UT dataset generated from Places2 and tested on test images from Places2 by GL, CA and GC. It shows that, after removing either the noise or the image stream, the performance degrades drastically. This implies that both the image and the noise pattern are crucial for extracting rich features for detection. The worst performance is observed when all 3 fusion modules removed, even though it still has the noise and the image streams. When adding either fusion module 1 and 2 or fusion module 3 back into the network, the performance is clearly improved. These results indicate that the proposed fusion module is essential for exchanging the information across multi-scale representations and achieving semantically richer feature fusion.

<table>
<thead>
<tr>
<th>Ablation of NIX-Net</th>
<th>Test mIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GL</td>
</tr>
<tr>
<td>w/o all fusion modules</td>
<td>79.36</td>
</tr>
<tr>
<td>w/o fusion module 1 and 2</td>
<td>84.72</td>
</tr>
<tr>
<td>w/o fusion module 3</td>
<td>89.19</td>
</tr>
<tr>
<td>Full NIX-Net</td>
<td>91.33</td>
</tr>
</tbody>
</table>

Table 2: Ablation of NIX-Net for (a) the noise/image stream or (2) the 3 multi-scale cross fusion modules. Networks are trained on the UT dataset generated for Places2 and tested on GL/CA/GC test sets.

6 Conclusion

In this work, we have proposed an effective approach for universal deep inpainting detection. Our approach consists of two important designs: 1) a novel universal training dataset generation method and 2) a Noise-Image Cross-fusion (NIX-Net) detection network. Extensive experiments on two benchmark datasets verify the effectiveness of our proposed approach and its superior generalization ability when applied to detect unseen deep inpainting methods. Our work not only provides a powerful universal detection method but also opens up a new direction for building more advanced universal deep inpainting detectors.
References


