2024 IEEE 12th International Conference on Computer Science and Network Technology

Photo Response Non-Uniformity Based AI-Generated Image Detection

Kaifeng Wu[1,2], Xiaolong Li[1,2]

[1] Institute of Information Science,Beijing Jiaotong University,Beijing,China

[2]Beijing Key Laboratory of Advanced Information Science and Network Technology,Beijing,China

Abstract

The hyper-realism of images generated by diffusion models has become increasingly impressive, making it difficult for humans to distinguish between AI-generated and real images. Most existing detectors rely on extracting artifacts inherent in the process of creating fake images, often ignoring the unique fingerprints left behind when real images are captured. It has been observed that real images exhibit a distinctive pattern noise known as Photo Response Non-Uniformity (PRNU). Based on this consideration, in this study, an AI-generated image detection method based on PRNU is proposed. Specifically, by combining PRNU features with original image features, we effectively capture the intrinsic noise patterns in real images and then train a classification network. Experimental results on 11 generative models demonstrate the practical effectiveness of this method in detecting AI-generated images.

Motivation

- The rapid development of AI-generated image technology in recent years, especially the rise of diffusion modeling, has led to an unprecedented level of realism in AI-generated images, and it is difficult for the human eye and traditional methods to distinguish between the generated image and the real image as shown in Figure 1.
- Most of the existing detection methods rely on identifying artifacts or specific patterns in the process of generating an image, but these methods tend to have weak generalization capabilities in the face of new generative models.

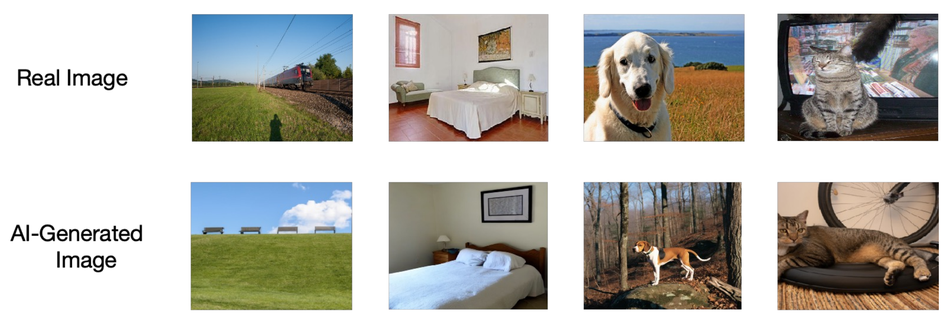

Figure 1 Real images and AI-generated images are hard to distinguish.

Methodology

To improve the generalization of AI-generated image detection, we focus on extracting general features from real images. As shown in Figure. 2, a detection framework is developed by combining the PRNU features with the original image features.

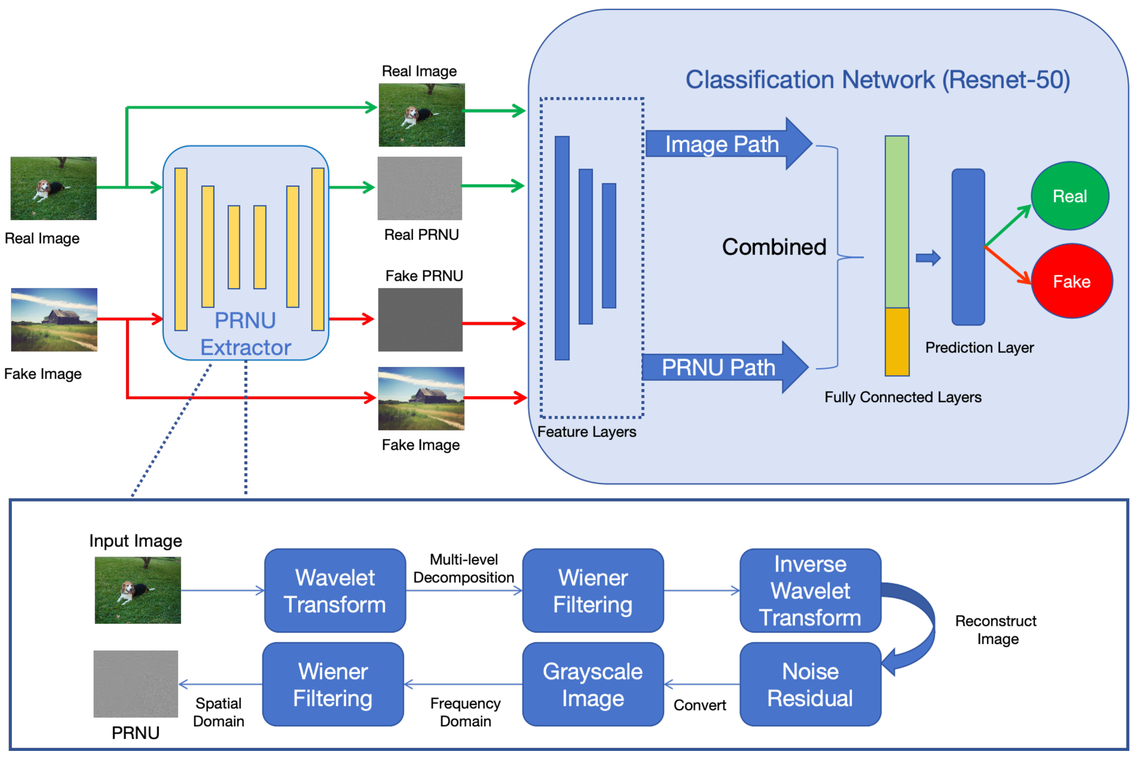

Figure 2 The proposed framework.

PRNU Extractor

1. First perform a four-level wavelet decomposition on each channel image I_c, decomposing the image into sub-bands of different frequency ranges,

$$\left\{C_A^j, C_H^j, C_V^j, C_D^j\right\}_{j=1}^{L} = DWT(I_c)$$

2. Adaptive wiener filtering to the high-frequency detail coefficients to remove non-uniform noise and preserve PRNU features,

$$C_D^{j'} = Wiener\left(C_D^j, \sigma^2\right)$$
$$C_H^{j'} = Wiener\left(C_H^j, \sigma^2\right)$$
$$C_V^{j'} = Wiener\left(C_V^j, \sigma^2\right)$$

3. The wavelet coefficients processed by Wiener filtering are then used to reconstruct the image by inverse wavelet transform, yielding the initial noise residual W_c, which retains the noise components associated with PRNU,

$$W_c = IDWT\left(C_A^L, C_H^{j'}, C_V^{j'}, C_D^{j'}\right)$$

4. The combined noise residual W is then zero mean normalized to remove the global mean while retaining the local PRNU features,

$$W_{zm} = W - \frac{1}{N}\sum_{i=1}^{N} W[i]$$

5. Fast Fourier Transform (FFT) is applied to the zero-mean normalized W_{zm} and Wiener filtering in the frequency domain,

$$W_{fft} = FFT(W_{zm})$$

$$W_{filtered} = W_{fft} \cdot \frac{\sigma^2}{\left|W_{fft}\right|^2 + \sigma^2}$$

6. Next, the inverse Fast Fourier Transform (IFFT) is performed to obtain the PRNU features after filtering in the frequency domain,

$$K = IFFT\left(W_{filtered}\right)$$

Experiments

Method→	CNNSpot [9]		FreDect [7]		Fusing [8]		Gram-Net [11]		DIRE-G [10]		UnivFD [18]		Ours	
Test Set	ACC	AP	ACC	AP	ACC	AP	ACC	AP	ACC	AP	ACC	AP	ACC	AP
IMLE	86.2	98.3	90.6	92.8	92.3	96.1	93.5	95.7	87.2	90.6	93.2	94.8	**94.4**	**99.3**
ProGAN	99.7	99.5	96.3	98.7	**100.0**	**100.0**	99.9	**100.0**	94.2	98.9	99.8	**100.0**	99.4	**100.0**
StyleGAN	**90.1**	**96.8**	76.5	88.6	82.7	96.8	85.6	**99.3**	83.0	91.4	84.7	96.3	78.4	87.7
BigGAN	71.1	84.6	81.3	92.5	76.3	87.2	67.3	90.6	70.1	76.4	**95.2**	**99.3**	75.1	81.3
StarGAN	94.6	99.0	94.3	99.5	96.2	99.8	94.9	99.2	95.4	99.3	95.3	99.1	**100.0**	**100.0**
VQDM	56.4	87.8	76.9	85.0	55.0	75.8	51.6	62.3	53.4	55.7	**85.2**	**96.3**	80.4	87.9
ADM	60.5	72.7	63.4	61.7	49.2	**93.8**	58.7	73.2	75.6	85.4	66.7	85.9	**78.2**	90.1
DALLE2	50.5	53.7	36.0	38.9	52.8	70.7	48.5	51.2	66.3	73.9	50.8	63.2	**79.3**	**97.3**
LDM	50.3	58.9	54.3	60.7	55.2	62.5	58.2	66.8	67.8	78.5	58.9	62.7	**91.6**	**99.2**
Glide	58.3	71.3	54.2	55.3	56.7	76.5	53.4	63.7	**70.2**	77.8	63.2	**82.9**	58.8	80.6
Midjourney	51.3	66.2	46.7	47.3	51.3	70.0	50.0	55.8	58.4	61.8	56.2	74.0	68.2	**91.2**
Avg(%)	76.0	78.9	70.0	74.6	69.8	82.9	66.1	75.8	74.3	80.9	73.8	82.7	**82.2**	**92.2**

Table 1 Evaluation results of different algorithms across on different test datasets. ACC indicates accuracy, AP indicates average precision. Bold values indicate best performance.

Detector	JPEG	Downsampling	Blur
CNNSpot [9]	64.0	60.3	67.2
FreDect [7]	70.3	35.9	72.5
Fusing [8]	62.3	52.8	67.3
GramNet [11]	64.2	57.8	66.9
DIRE-G [10]	63.4	56.2	64.1
UnivFD [18]	76.9	77.2	75.8
Ours	**78.1**	**79.1**	**77.4**

Table 2 Detection accuracy (average of various test datasets) over distorted images. the black body means the best.