COMPUTER-GENERATED IMAGE FORENSICS BASED ON VISION TRANSFORMER WITH HIGH-FREQUENCY FEATURE ENHANCEMENT

Yifang Chen
School of Cyber Security,
Guangdong Polytechnic Normal University
Guangzhou, China
chenyf@gpnu.edu.cn

Tao Mai
School of Cyber Security,
Guangdong Polytechnic Normal University
Guangzhou, China
thomasmak@stu.gpnu.edu.cn

Shanshan Wang
School of Cyber Security,
Guangdong Polytechnic Normal University
Guangzhou, China
shanshanwang@stu.gpnu.edu.cn

Xiangui Kang
Guangdong Key Laboratory of Information Security,
Sun Yat-sen University
Guangzhou, China
isskxg@mail.sysu.edu.cn

***Abstract*—Distinguishing computer-generated (CG) images from photographic (PG) images is an important task in multimedia forensics. Many deep learning-based methods have recently been proposed for CG image forensics. However, the detection performances of these methods still need to be improved, especially in terms of robustness against post-processing operations, thus limiting their practical applicability. To tackle these issues, we leverage the *Vision Transformer* (ViT) model, which excels in capturing the global features of images, and design a High-Frequency Feature Enhancement (HFFE) module to exploit the discriminative frequency information between CG and PG images. In our experiments, we evaluate the performance under various commonly used post-processing operations. Moreover, we test the performance in the presence of adversarial attacks, which is a more challenging real-world case. The experimental results demonstrate that our method achieves superior detection accuracy and significantly better robustness against post-processing operations and adversarial attacks when compared with the state-of-the-art methods.**

***Index Terms*—Computer-generated images, robustness, adversarial attacks, vision transformer, high-frequency feature enhancement.**

I. Introduction

NOWADAYS, computer-generated (CG) images, which are often generated by using computer graphics techniques (e.g., 3D rendering techniques [1], [2]) or advanced deep learning algorithms such as autoencoders (AE) [3], [4] and Generative Adversarial Networks (GANs) [5], [6], are difficult to recognize with the naked eye and may present potential risks to social stability if used maliciously. Moreover, in practical scenarios, the CG images transmitted over the Internet may undergo post-processing operations such as compression and resizing, which challenge the robustness of CG image detection. Therefore, it is of primary importance to develop robust methods to distinguish CG images from photographic (PG) images.

In recent years, deep neural networks, such as Convolutional Neural Networks (CNNs), have been successfully used for CG image forensics due to their powerful learning ability [7], [8]. Bai et al. [9] contributed a Large-Scale CG images Benchmark (LSCGB) and further proposed a texture-aware network to distinguish CG and PG images. Yao et al. [10] developed a CG image detection method by utilizing transfer learning and convolutional attention. Meena et al. [11] proposed a two-stream network that utilizes RGB color features and high-frequency noise features obtained by Steganalysis Rich Model (SRM) filters [12]. Gangan et al. [13] employed Multi-Colorspace and EfficientNet [14] for the task of detecting CG images. Chen et al. [15] designed a forensics contrastive learning framework to adaptively learn intrinsic forensics features for the detection of CG images.

Despite advances in CNN-based approaches for CG image forensics, their performance will be degraded greatly when detecting post-processed CG images, limiting their practical applicability in the real world. In addition, limited research has been done to address the more challenging practical scenario, i.e., detecting the CG images in the presence of adversarial attacks. In CNN-based approaches, the inherent characteristic of limited receptive fields results in an overemphasis on local features such as texture and edges. Since all the regions of a CG image are synthesized, a wide range of artifacts that span

This work was supported in part by the National Natural Science Foundation of China under Grant no. 62102100/62102462, the Basic and Applied Basic Research Foundation of Guangdong Province under Grant no. 2022A1515010108, the Opening Project of Guangdong Provincial Key Laboratory of Information Security Technology under Grant no. 2023B1212060026, Key Research Platforms and Projects of Universities in Guangdong Province Grant no. 2024ZDZX1038, and Research Project of Guangdong Polytechnic Normal University under Grant no. 2021SDKYA127/2022SDKYA027.

the entire image can be created in the computer generation process. Therefore, the global features are also crucial in CG image forensics for providing essential information regarding the artifacts of generation. Vision Transformer (ViT) [16] has recently emerged as a competitive alternative to CNNs and has increasingly been applied to the image forensics tasks, such as the detection of splicing [17], deepfakes [18], and recaptured screen images [19], etc. Compared with CNNs, the cascaded self-attention modules in ViT can capture long-range feature dependencies and reflect complex spatial transformations to capture the global features. Furthermore, while CNNs exhibit vulnerability to adversarial attacks, ViT demonstrates better robustness against adversarial attacks [20]–[22].

In this work, we propose a ViT with high-frequency feature enhancement for CG image forensics. We design a High-Frequency Feature Enhancement (HFFE) module to exploit the high-frequency features, which are one of the key features to differentiate CG images from PG images. The input images are first processed by the HFFE module and then fed to the ViT module. The HFFE module mainly comprises a convolutional block and a high-frequency feature extractor, which can simultaneously extract distinct local features and frequency features from the input images. The high-frequency feature extractor is designed to successively process the images through Fast Fourier Transform (FFT), Gaussian high-pass filtering, and Inverse Fast Fourier Transform (IFFT). In Fig. 1, we show the high-frequency features obtained from original and post-processed images by SRM filters used by Meena et al. [11] and the proposed high-frequency feature extractor, respectively. It can be seen that the features extracted by SRM filters are more sensitive to post-processing operations, while the proposed extractor achieves better robustness.

Our main contributions are as follows: (1) A novel ViT with High-Frequency Feature Enhancement (HFFE) module is proposed for robust CG image forensics. (2) We consider a challenging practical scenario in which CG image detection is conducted in the presence of adversarial attacks. (3) Experimental results demonstrate that the proposed method achieves strong robustness against post-processing operations and adversarial attacks.

II. Proposed Method

A. Overall network architecture

The architecture of our proposed method is shown in Fig. 2. Firstly, the input images go through the HFFE module to be converted into feature map patches. These patches are then summed, flattened, and mapped to a series of token embeddings. Then, the token embeddings pass through the transformer encoder. Additionally, instead of using the classification (CLS) token to gain the classification features, we use global average pooling which is frequently employed to integrate visual features from different spatial locations to guarantee translation invariance. Finally, after global average pooling, the resulting output from the transformer encoder is fed to the classifier. In this work, the ViT-B/16 model [16] serves as the baseline model.

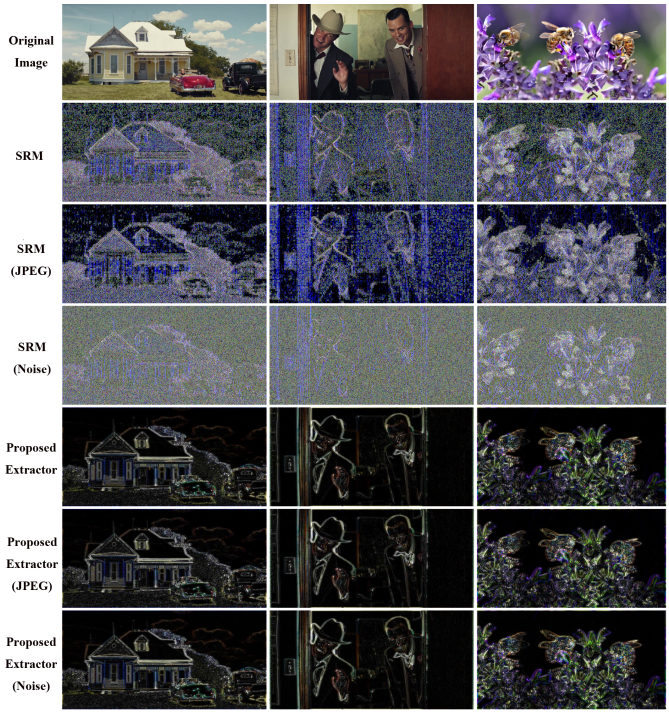

Fig. 1. Images taken from the LSCGB [9], as well as their high-frequency features obtained through SRM filters and the proposed high-frequency feature extractor under different post-processing operations, namely JPEG compression (quality factor (QF) = 50) and Gaussian noise addition (zero mean and σ = 1), with triple brightness.

B. High-Frequency Feature Enhancement

In order to fully mine the discriminative operties between CG and PG images and improve the robustness of the detector, we designed the HFFE module. As shown in Fig. 2, the HFFE module comprises a convolutional block for extracting local feature maps, a high-frequency feature extractor for extracting high-frequency feature maps, and two convolutional layers for converting these feature maps into patches.

In the convolutional block, we leverage the advantage of convolution operations to extract local features, because ViT is not as proficient as CNNs in capturing local features such as texture and edges in shallow layers [23]. These local features also contribute to CG image forensics.

The convolutional block consists of a convolutional layer, a batch normalization layer, and a maximum pooling layer. For the input image $x \in R^{H \times W \times 3}$, the output of the convolutional block can be denoted as:

$$x_l = MaxPool\left(BN\left(Conv1\left(x\right)\right)\right) \tag{1}$$

where $x_l \in R^{I \times J \times C}$, (H,W) is the size of the input image, (I,J) is the size of the output of the convolutional block, and C is the number of channels.

We design a high-frequency feature extractor to extract robust high-frequency features. As shown in Fig. 2, an input image is split into three color channels, i.e., R, G, and B. For k_{th} color channel $f^k\left(x,y\right)$ of size $H \times W$, the process of extracting high-frequency feature is as follows: First, $f^k\left(x,y\right)$

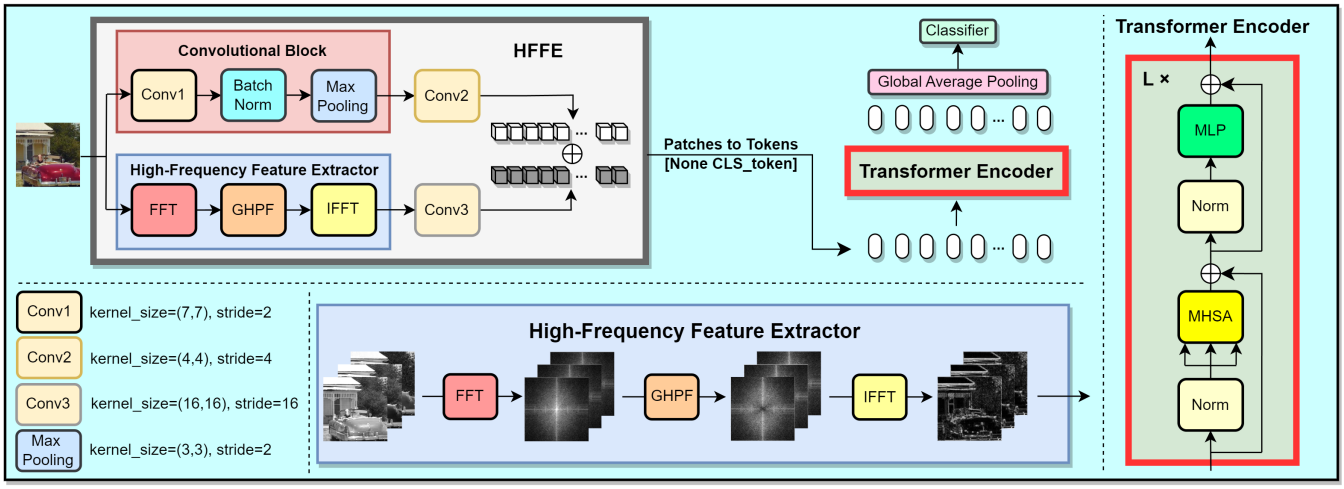

Fig. 2. The network architecture of the proposed method. "⊕" represents an addition, "L" represents the number of blocks and we set its value to twelve.

is transformed from the spatial domain to the frequency domain by using Fast Fourier Transform (FFT). It can be noted as:

$$F^k(u,v) = \sum_{x=0}^{H-1}\sum_{y=0}^{W-1} f^k(x,y)\cdot e^{-i2\pi\left(\frac{ux}{H}+\frac{vy}{W}\right)} \tag{2}$$

where (x, y) and (u, v) are the coordinates of the image in the spatial domain and frequency domain, respectively. Second, a Gaussian high-pass filter (GHPF) formulated as Eq. 3 is applied to the image for filtering in the frequency domain.

$$H(u,v) = 1 - e^{-\frac{D^2(u,v)}{2{D_0}^2}} \tag{3}$$

where $D_0 \in R$ is the cut-off frequency and $D(u, v)$ is the distance from the frequency point (u, v) to the center of the spectrum. The filtering result can be noted as:

$$G^k(u,v) = F^k(u,v)\cdot H(u,v) \tag{4}$$

Finally, the frequency domain information is transformed back to the spatial domain information by using Inverse Fast Fourier Transform (IFFT). It can be noted as:

$$x_h^k(x,y) = \frac{1}{HW}\sum_{u=0}^{H-1}\sum_{v=0}^{W-1} G^k(u,v)\cdot e^{i2\pi\left(\frac{ux}{H}+\frac{vy}{W}\right)} \tag{5}$$

The high-frequency features are extracted from each color channel and the final output is $x_h \in R^{H\times W\times 3}$.

Both the output of convolutional block x_l and the output of high-frequency feature extractor x_h are split into patches of size (P, P) and added together to the new patches $x_p \in R^{\frac{H}{P}\times\frac{W}{P}\times\left(P^2\times 3\right)}$. It can be noted as:

$$x_p = Conv2(x_l) + Conv3(x_h) \tag{6}$$

Then the feature map patches x_p are flattened and mapped to a series of token embeddings $x_t \in R^{N\times D}$, where $N = HW/P^2$ and $D = P^2 \times 3$ are the number and the size of token embeddings, respectively.

C. Transformer Encoder

The transformer encoder consists of twelve stacked ViT blocks, where each block comprises two sub-layers: Multi-Head Self-Attention (MHSA) and a Feed-Forward Network (FFN), also referred to as a Multi-Layer Perceptron (MLP). Layer normalization (LN) [24] is applied before each sub-layer, with a residual connection surrounding them. In the MHSA layer, token embeddings $x_t \in R^{N\times D}$ are linearly transformed into qkv spaces (i.e., queries $Q \in R^{N\times D}$, keys $K \in R^{N\times D}$, and values $V \in R^{N\times D}$). The token embeddings are split and fed to self-attention modules for twelve executions in parallel. The resulting outputs of the self-attention modules are concatenated and projected. In the MLP layer, element-wise operations are performed, which are applied individually to each token. Specifically, it first expands the embedding dimension from 768 to 3072, followed by a non-linear activation GELU [25], and then projects it back to 768. For the MHSA, by calculating the dot product, the similarity between different tokens can be calculated to obtain long-range and global attention. The corresponding values of V are linearly aggregated. For the MLP, each token is performed dimension alteration and non-linear transformation, thereby enhancing the representation ability of the token.

III. Experiments

A. Experiment Setup

The benchmark database used in this study is the LSCGB proposed by Bai et al. [9], which is the state-of-the-art database for CG image forensics. The LSCGB contains 71,168 CG images and 71,168 PG images. All images are randomly divided into training set, testing set, and validation set according to the same ratio in [9] to 7:1:2. The input images are conducted the same processing as the method in [9]. The experiments are carried out using PyTorch library on a single NVIDIA GTX3090. The total number of training epochs is set to 50. The Adam [26] is used as the optimizer, and the batch size is

set to 32. For CNN-based methods and our ViT-based method, the learning rate is initialized to 0.0001 and 0.005 respectively, and scheduled to decrease by 10% every five epochs.

B. Evaluation of Robustness

In this section, we evaluate the robustness of our proposed method against various post-processing operations and adversarial attacks. We consider four common post-processing operations with different parameters: JPEG compression (quality factor (QF) $\in \{90, 80, 70\}$), image scaling (up by 20% or down by 20%), image blur (median blur and mean blur, kernel size $\in \{3 \times 3\}$), and Gaussian noise addition (zero mean and $\sigma \in \{1, 1.5\}$). We also consider four common types of black-box adversarial attacks, including ST [27] (the translation in any direction as a percentage of the image size: $p \in \{5\%, 10\%\}$), HSJA [28] (iterations $i \in \{25, 50\}$), SimBA [29] (pixel-based, 100 iterations and $\varepsilon \in \{0.5, 1.0\}$), SA [30] (100 iterations and $\varepsilon \in \{5, 10\}$).

We compare our method with the state-of-the-art methods [9] [10] [11] [15]and ViT [16]. The testing results are reported in Table I. It can be observed that the basic ViT can achieve satisfying performance and our proposed method further improves the performance. Our method achieves an accuracy of 95.63% on the original testing dataset, which is 0.62% higher than Chen et al. [15] and approximately 5% higher than the other methods. Under various post-processing operations and black-box attacks, our method has an average accuracy close to 90%, which outperforms others in the comparison by more than 10%. Specifically, compared to the accuracy on the original dataset, our method shows an average accuracy decrease of 3.43%, 0.80%, 6.62%, and 0.87% under four types of post-processing operations respectively. In comparison, the best-performing method among others, as demonstrated by Chen et al. [15], suffers a larger decrease in average accuracy of 12.38%, 3.14%, 7.15%, and 5.96% under the same conditions. Our method achieved robust performance in all four post-processing operations. The performances of our method rank second under Mean Blur post-processing and are lower than Chen et al. [15]. This may be because Chen et al. [15] use data augmentation in their training, resulting in better performance in some post-processing. However, our method still leads the way in all post-processing robust tests Furthermore, our method yields a decrease of 3.46%, 9.64%, 3.50%, and 4.13% under four types of black-box attacks. Meanwhile, the method proposed by Bai et al. [9], shows a decrease of 13.22%, 22.27%, 21.25%, and 13.77%. These results demonstrate the superior robustness of our method against post-processing operations and adversarial attacks compared with other CNN-based thods. It is noted that our method demonstrates better robustness against adversarial attacks compared to ViT. This could be because the images are first converted into feature maps in our method, rather than inputting image patches in conventional ViT. Thus, our method is relatively insensitive to adversarial attacks which target pixel-level perturbations on the image.

TABLE I
THE DETECTION ACCURACY UNDER POST-PROCESSING OPERATIONS AND ADVERSARIAL ATTACKS

Methods⟶ Attacks ↓	Bai [9]	Yao [10]	Meena [11]	Chen [15]	ViT [16]	Ours
Origin.	91.73	91.26	90.82	95.01	94.88	**95.63**
JPEG QF=90	84.28	83.17	82.69	86.33	90.76	**93.61**
JPEG QF=80	78.89	78.29	76.84	82.20	88.31	**92.12**
JPEG QF=70	76.24	75.91	74.57	80.71	86.45	**90.87**
Scaling Up 20%	89.36	87.78	87.34	94.01	93.98	**95.06**
Scaling Down 20%	88.04	88.16	86.42	89.18	93.45	**94.61**
Median 3×3	71.74	70.88	70.45	88.96	88.13	**89.59**
Mean 3×3	67.83	66.39	65.83	**94.44**	87.59	88.43
Noise σ=1	84.55	82.73	81.92	89.59	93.78	**95.09**
Noise σ=1.5	82.64	81.71	81.37	88.51	93.43	**94.43**
ST p=5%	83.59	82.42	79.74	81.25	91.52	**93.11**
ST p=10%	79.27	78.45	75.91	77.34	89.16	**91.23**
HSJA i=25	75.65	76.43	74.32	75.75	87.18	**89.29**
HSJA i=50	68.57	69.91	67.71	68.42	80.54	**82.68**
SimBA ε=0.5	74.94	72.70	71.38	82.89	88.79	**92.89**
SimBA ε=1.0	72.31	69.75	68.92	79.57	87.43	**91.38**
SA ε=5	81.57	80.73	79.25	62.16	91.01	**92.77**
SA ε=10	80.65	78.86	77.91	59.38	87.71	**90.23**

TABLE II
ABLATION EXPERIMENTS ON THE PROPOSED METHOD

Methods⟶ Scenarios ↓	ViT (Baseline)	w/o High-Freq.	w/o Conv. Block	Ours
Origin.	94.88	95.42	95.36	**95.63**
JPEG QF=90	90.76	91.20	92.79	**93.61**
JPEG QF=80	88.31	89.56	91.65	**92.12**
JPEG QF=70	86.45	88.74	89.82	**90.87**
SimBA ε=0.5	88.79	89.38	92.43	**92.89**
SimBA ε=1.0	87.43	87.94	91.17	**91.38**

C. Ablation Study

In this section, we assess the convolutional block (Conv. Block) and the high-frequency feature extractor (High-Freq.) in terms of robustness against post-processing operations and adversarial attacks. We test their performances on the original dataset, the dataset edited by JPEG compression (quality factor (QF) $\in \{90, 80, 70\}$) and SimBA [29] (pixel-based, 100 iterations and $\varepsilon \in \{0.5, 1.0\}$), respectively.

As shown in Table II, the accuracy of the original dataset decreases without the high-frequency feature extractor or the convolutional block. Without the high-frequency feature extractor, the average accuracy declines by 2.37% and 3.50% under JPEG compression and SimBA [30], respectively. Similarly, without the convolutional block, the average accuracy witnessed a decrease of 0.78% and 0.34% under the same conditions, respectively. The performance degradation confirms that utilizing the high-frequency feature extractor or convolutional block effectively improves the model's performance.

Moreover, we apply t-SNE [31] to visualize the feature distribution of both the baseline (i.e., ViT) and our model in several experimental scenarios. As shown in Fig. 3, our model significantly minimizes the overlap area between the features of CG and PG images in all experimental scenarios when compared with the baseline. The visualization of these reduced-dimensional features further supports the superiority of our model.

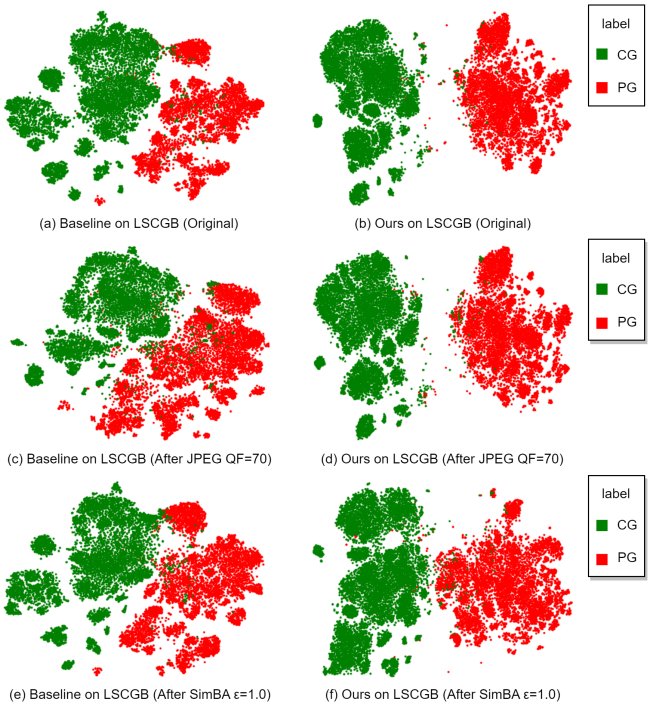

Fig. 3. T-SNE feature distribution visualizations of baseline and our model.

IV. Conclusion

In this work, we propose a novel ViT with High-Frequency Feature Enhancement (HFFE) module for CG image forensics. The advantage of ViT in capturing global features contributes to distinguishing CG images from PG images, and the HFFE module which exploits the discriminative frequency information further improves the detection performance. Extensive experiments have shown that our method outperforms state-of-the-art methods, especially in terms of robustness against post-processing operations and adversarial attacks. In further work, the proposed framework will also be extended and modified to tackle more image forensics applications, such as image tampering detection.

References

[1] H. Shum and S. B. Kang, "Review of image-based rendering techniques," in *Proc. SPIE*, vol. 4067, pp. 2–13, 2000.

[2] P. Goswami, "A survey of modeling, rendering and animation of clouds in computer graphics," *Vis. Comput.*, vol. 37, no. 7, pp. 1931–1948, 2021.

[3] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[4] H. Huang, Z. li, R. He, Z. Sun, and T. Tan, "IntroVAE: Introspective variational autoencoders for photographic image synthesis," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, vol. 31, pp. 52–63, 2018.

[5] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," *IEEE Signal Process. Mag.*, vol. 35, no. 1, pp. 53–65, 2018.

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[7] R. Huang, F. Fang, H. H. Nguyen, J. Yamagishi, and I. Echizen, "A method for identifying origin of digital images using a convolutional neural network," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA ASC)*, pp. 1293–1299, 2020.

[8] R.-S. Zhang, W.-Z. Quan, L.-B. Fan, L.-M. Hu, and D.-M. Yan, "Distinguishing computer-generated images from natural images using channel and pixel correlation," *J. Comput. Sci. Technol.*, vol. 35, pp. 592–602, 2020.

[9] W. Bai, Z. Zhang, B. Li, P. Wang, Y. Li, C. Zhang, and W. Hu, "Robust texture-aware computer-generated image forensic: Benchmark and algorithm," *IEEE Trans. Image Process.*, vol. 30, pp. 8439–8453, 2021.

[10] Y. Yao, Z. Zhang, X. Ni, Z. Shen, L. Chen, and D. Xu, "CGNet: Detecting computer-generated images based on transfer learning with attention module," *Signal Process. Image Commun.*, vol. 105, p. 116692, 2022.

[11] K. B. Meena and V. Tyagi, "Distinguishing computer-generated images from photographic images using two-stream convolutional neural network," *Appl. Soft Comput.*, vol. 100, p. 107025, 2021.

[12] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Trans. Inf. Forensics Secur.*, vol. 7, no. 3, pp. 868–882, 2012.

[13] M. P. Gangan, K. Anoop, and V. Lajish, "Distinguishing natural and computer generated images using Multi-Colorspace fused EfficientNet," *J. Inf. Secur. Appl.*, vol. 68, p. 103261, 2022.

[14] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, pp. 6105–6114, 2019.

[15] Y. Chen, W. Yin, A. Luo, J. Yang, and J. Wang, "Improving the generalization and robustness of computer-generated image detection based on contrastive learning," *Int. J. Intell. Syst.*, vol. 2025, no. 1, p. 9939096, 2025.

[16] A. Dosovitskiy *et al.*, "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Representations*, 2020.

[17] Y. Sun, R. Ni, and Y. Zhao, "ET: Edge-enhanced transformer for image splicing detection," *IEEE Signal Process. Lett.*, vol. 29, pp. 1232–1236, 2022.

[18] Y.-J. Heo, W.-H. Yeo, and B.-G. Kim, "Deepfake detection algorithm based on improved vision transformer," *Appl. Intell.*, vol. 53, no. 7, pp. 7512–7527, 2023.

[19] G. Li, H. Yao, Y. Le, and C. Qin, "Recaptured screen image identification based on vision transformer," *J. Vis. Commun. Image Represent*, vol. 90, p. 103692, 2023.

[20] N. Carlini and H. Farid, "Evading deepfake-image detectors with white- and black-box attacks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, pp. 658–659, 2020.

[21] K. Mahmood, R. Mahmood, and M. Van Dijk, "On the robustness of vision transformers to adversarial examples," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, pp. 7838–7847, 2021.

[22] S. Paul and P.-Y. Chen, "Vision transformers are robust learners," in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, pp. 2071–2081, 2022.

[23] X. Mao, G. Qi, Y. Chen, X. Li, R. Duan, S. Ye, Y. He, and H. Xue, "Towards robust vision transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pp. 12042–12051, 2022.

[24] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[25] D. Hendrycks and K. Gimpel, "Gaussian error linear units (GELUs)," *arXiv preprint arXiv:1606.08415*, 2016.

[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[27] L. Engstrom, B. Tran, D. Tsipras, L. Schmidt, and A. Madry, "Exploring the landscape of spatial robustness," in *Proc. Int. Conf. Mach. Learn.*, pp. 1802–1811, 2019.

[28] J. Chen, M. I. Jordan, and M. J. Wainwright, "Hopskipjumpattack: A query-efficient decision-based attack," in *Proc. IEEE Symp. Secur. Privacy (SP)*, pp. 1277–1294, 2020.

[29] C. Guo, J. Gardner, Y. You, A. G. Wilson, and K. Weinberger, "Simple black-box adversarial attacks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pp. 2484–2493, 2019.

[30] M. Andriushchenko, F. Croce, N. Flammarion, and M. Hein, "Square attack: a query-efficient black-box adversarial attack via random search," in *Proc. Eur. Conf. Comput. Vis.*, pp. 484–501, 2020.

[31] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.