Computer-Generated Image Forensics Based on Vision Transformer with Forensic Feature Pre-processing Module

Yifang Chen, Guanchen Wen, Yong Wang, Jianhua Yang(✉), and Yu Zhang

Guangdong Polytechnic Normal University, GuangZhou 510665, GuangDong, China
yangjh86@gpnu.edu.cn

Abstract. The correct distinction between highly realistic computer-generated (CG) images and photographic (PG) images has become an important area of research. In recent years, most of the CG image forensics methods are proposed based on deep learning, but the detection performances of these methods still need to be improved, especially in terms of robustness and generalization. To tackle these issues, we leverage the *Vision Transformer* (ViT) model, which excels in capturing the global features of images, and design a Forensic Feature Pre-processing (FFP) module to further improve the detection performance. Experiments are conducted on a large-scale CG image benchmark (LSCGB), which is a challenging dataset for CG image detection. The proposed approach can achieve high detection accuracy. Extensive experiments on different public datasets and common post-processing operations demonstrate our approach can achieve significantly better generalization and robustness than the state-of-the-art approaches.

Keywords: Computer-generated images · Vision Transformer · Robustness · Generalization

1 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. Therefore, it is of primary importance to develop reliable methods to distinguish CG images from photographic (PG) images, which are captured by digital cameras and accurately and objectively record real-world scenes.

In recent years, deep neural networks, such as Convolutional Neural Networks (CNNs), have been successfully used for CG image forensics due to their powerful

J. Cai et al. (Eds.): SPNCE 2023, LNICST 525, pp. 315–325, 2025.
https://doi.org/10.1007/978-3-031-73699-5_22

learning ability [7–9]. Bai et al. [10] contributed a Large-Scale CG images Benchmark (LSCGB). They further proposed a texture-aware network to serve as a strong baseline for the new benchmark based on the observation that the texture feature is an effective representation to distinguish CG and PG images. Yao et al. [11] proposed a CG image detection method by applying transfer learning and convolutional attention to consider both the shallow content features and the deep semantic features of the image, thereby improving the accuracy of identifying CG images. Gangan et al. [12] proposed a Multi-Colorspace fused EfficientNet [13] model by parallelly fusing three EfficientNet networks. Each of the three networks operates in a different colorspace, i.e., RGB, HSV, and LCH to provide high classification accuracies for the task of detecting CG images. Meena et al. [14] found that the high-frequency noise features of CG images are significantly different from those of PG images, so they can be utilized as complementary discriminate features to improve detection performance. Therefore, they proposed a two-stream network to respectively extract RGB color features and high-frequency noise features obtained from Spatial Rich Model (SRM) [15] filters.

Despite CNNs having been proven effective tools for CG image forensics, they tend to learn local visual discriminative features (e.g., color, texture, and edges) of the images and fail to capture global correlation among different image regions due to the limitation of receptive fields. Since all the regions of a CG image are synthesized, a wide range of artifacts that span the entire image can be created in the computer-generated process. Therefore, these global features are also crucial in CG image forensics for providing essential information regarding the artifacts of generation. Vision Transformer (ViT) [16] have recently emerged as a competitive alternative to CNNs and increasingly be 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 help it to capture long-range feature dependencies and reflect complex spatial transformations to capture the global features.

In this work, we apply ViT specifically to CG image forensics and design it to further improve its detection performance. Instead of inputting image patches in conventional ViT, we propose a Forensic Feature Pre-processing (FFP) module to first convert the input images to feature maps which is beneficial to CG image forensics. The FFP module mainly comprises a convolutional block and a SRM filter block, which extract distinct spatial features and noise domain features from the input images, respectively. In the convolutional block, the input images are successively passed to a convolutional layer, a batch normalization layer, and a maximum pooling layer. In the SRM filter block, the input images are split into three color channels, i.e., R, G, and B, and then processed by three different SRM filters, respectively. The SRM filters can learn noise-based distinct features which have been proven to be effective for CG image detection in [14]. In Fig. 1, we show the features obtained from the images taken from LSCGB [10] by the three SRM filters. It can be seen that the SRM filters tend to suppress image content and focus on the local noise features of the images.

Our main contributions are as follows: (1) We specifically adapt ViT for CG image forensics. The images are first converted into discriminative features and then inputted into the ViT, rather than inputting image patches in conventional ViT. (2) A Forensic Feature Pre-processing (FFP) module is propped to highlight the discriminative information between CG and PG images. (3) Experimental results demonstrate that the proposed method achieves strong robustness against post-processing operations and generalization on different CG image datasets.

Fig. 1. The top-to-bottom images are respectively the original images taken from LSCGB [10] and the images of their RGB channels after passing through SRM filters.

2 Proposed Method

2.1 Overall Network Architecture

The architecture of the proposed ViT is shown in Fig. 2. Firstly, the input images go through the FFP module to be converted into feature map patches. These patches are then flattened and mapped to a sequence of token embeddings. These token embeddings are then combined with an additional class token, and the resulting embeddings are fed into a series of stacked transformer encoder blocks. Finally, the resulting output is input to the classifier for classification. In this work, the ViT-B/16 model [16] serves as the baseline model.

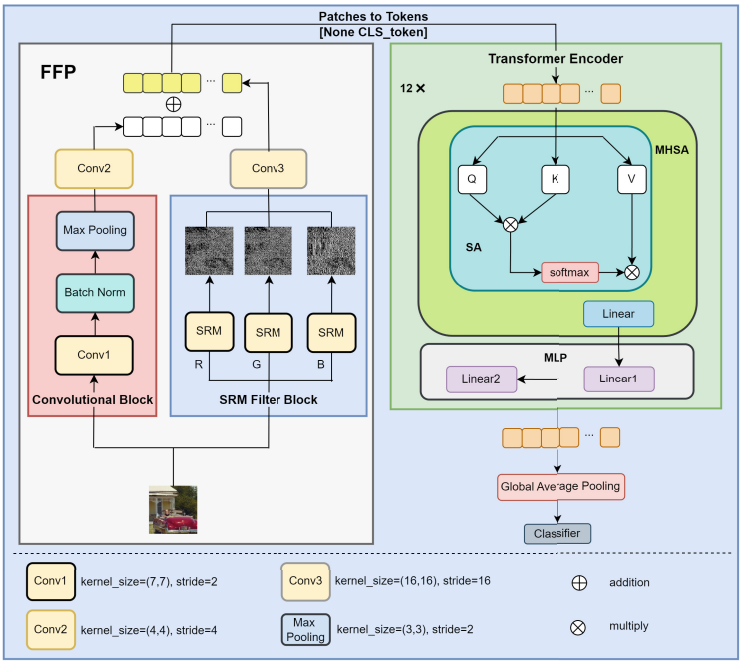

Fig. 2. The network architecture of the proposed method.

In order to further improve the detection performance of ViT, we made some improvements to it. Firstly, conventional ViT directly labeled patches with the fixed size from the original input image. This simple recognition makes it difficult for the model to extract local features of some basic structures in the image [20,21]. However, local features also contribute to CG image forensics. In order to take full advantage of local features and global features, we leverage the advantage of CNNs in extracting local features from images by using a convolutional block to extract local features. Secondly, local noise is one of the key features that can be used to differentiate CG images from PG images [14]. Therefore, we introduce a SRM filter block to extract the noise features which provide complementary clues for CG image detection.

Additionally, CNNs frequently use global average pooling layers before the final classifier to integrate visual features from different spatial locations to guarantee translation invariance (i.e. the network's predicted class for the object in the image isn't changed with the translation of their position), which is an important property of CNN. However, ViT differs from CNN in that it uses an additional class (CLS) token for performing classification, rather than relying on translation invariance. Based on the above mentioned, we use global average pooling to gain the classification features instead of the CLS token.

2.2 Forensic Feature Pre-processing Module

In order to fully mine the discriminative properties between CG and PG images, we designed the Forensic Feature Pre-processing (FFP) module. As shown in Fig. 2, the FFP module comprises a convolutional block for extracting local feature maps, a SRM filter block for extracting noise-based 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 [21]. 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.

The SRM filter block is applied to extract noise-based discriminate 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 x^k of size $H \times W$, one SRM filter F^k is used to extract the local noise feature x_h^k

$$x_h^k = F^k\left(x^k\right) \tag{2}$$

where $k = 1, 2, 3$ is the number of color channels. Figure 3 shows the weights of three SRM filters used in the SRM filter block. For each color channel, the size of local noise feature maps x_h^k can be calculated as follows:

$$\left\lfloor \frac{H + 2p - k}{s} + 1 \right\rfloor \cdot \left\lfloor \frac{W + 2p - k}{s} + 1 \right\rfloor \tag{3}$$

where p represents padding, s represents stride, (k, k) represents the filter size and $\lfloor \cdot \rfloor$ represents the floor function. In this work, to maintain consistency between the image through the SRM filter and the original image, we set (k, k) as $(5, 5)$, s as 1, and p as 2. The noise-based discriminate features are extracted from each color channel and the final output is x_h.

$$\frac{1}{4}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 2 & 4 & 2 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \frac{1}{12}\begin{bmatrix} -1 & 2 & -2 & 2 & -1 \\ 2 & -6 & 8 & -6 & 2 \\ -2 & 8 & -12 & 8 & -2 \\ 2 & -6 & 8 & -6 & 2 \\ -1 & 2 & -2 & 2 & -1 \end{bmatrix} \quad \frac{1}{2}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Fig. 3. The weights of three SRM filters F^1, F^2 and F^3.

Both the output of convolutional block x_l and the output of SRM filter block 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:

$$x_p = Conv2\left(x_l\right) + Conv3\left(x_h\right) \tag{4}$$

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.

2.3 Transformer Encoder

The transformer encoder consists of twelve stacked ViT blocks, where each block comprises two sub-layers: Multi-Head Self-Attention (MHSA) and Multi-Layer Perceptron (MLP). Layer normalization (LN) [22] is applied prior to each sub-layer, with a residual connection surrounding them.

In the MHSA layer, token embeddings are linearly transformed into qkv (i.e., queries $Q \in R^{N \times D}$, keys $K \in R^{N \times D}$, and values $V \in R^{N \times D}$) spaces, which are split and fed to self-attention (SA) modules for twelve executions in parallel. The resulting outputs are concatenated and projected. The SA module can be noted as:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{D}}\right)V \tag{5}$$

In the MLP layer, the element-wise operations are performed, which are applied individually to each token. It comprises two linear transformations, separated by a non-linear activation GELU [23]. It can be noted as:

$$MLP(x) = Linear2\left(GELU\left(Linear1\left(x\right)\right)\right) \tag{6}$$

For the MHSA, by calculating the dot product, the similarity between different tokens can be calculated to obtain long-range and global attention. And 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.

3 Experiments

3.1 Experiment Setup

The benchmark database used in this study is the LSCGB proposed by Bai et al. [10], 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 [10] to 7:1:2. The input images are conducted the same processing in [10]. 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 [24] 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.

3.2 Evaluation of Robustness

In this section, we evaluate the robustness of our proposed method against various post-processing operations. 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 compare our method with the state-of-the-art methods mentioned above and ViT. The testing results are reported in Table 1. 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 95.55% on the original testing dataset, which is approximately 5% higher than other methods. Under various post-processing operations, 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 5.27%, 1.48%, 7.37%, and 1.50% under four types of post-processing operations respectively. In comparison, the best-performing CNN-based method among others, as demonstrated by Bai et al. [10], suffers a larger decrease in average accuracy of 11.93%, 3.03%, 21.95%, and 8.14% under the same conditions.

As shown in Fig. 4, the detection of CG images becomes increasingly challenging as the intensity of the post-processing operation increases. The performance of other state-of-the-art methods declines sharply, particularly for images that have undergone median filtering. In contrast, our method maintains a high level of detection accuracy in all scenarios. These results demonstrate the superior robustness of our method against post-processing operations compared with other CNN-based methods.

Table 1. The detection accuracy under post-processing operations

Methods⟶ Operations ↓	Bai [10]	Yao [11]	Gangan [12]	Meena [14]	ViT [16]	Ours
Origin.	91.73	91.26	90.56	90.82	94.88	**95.55**
JPEG QF = 90	84.28	83.17	82.24	82.69	90.76	**91.43**
JPEG QF = 80	78.89	78.29	76.68	76.84	88.31	**90.29**
JPEG QF = 70	76.24	75.91	73.45	74.57	86.45	**89.13**
Scaling Up 20%	89.36	87.78	86.94	87.34	93.98	**94.31**
Scaling Down 20%	88.04	88.16	85.32	86.42	93.45	**93.84**
Median 3×3	71.74	70.88	69.93	70.45	88.13	**88.61**
Mean 3×3	67.83	66.39	64.74	65.83	87.59	**87.75**
Noise $\sigma = 1$	84.55	82.73	81.17	81.92	93.78	**94.36**
Noise $\sigma = 1.5$	82.64	81.71	80.08	81.37	93.43	**93.75**

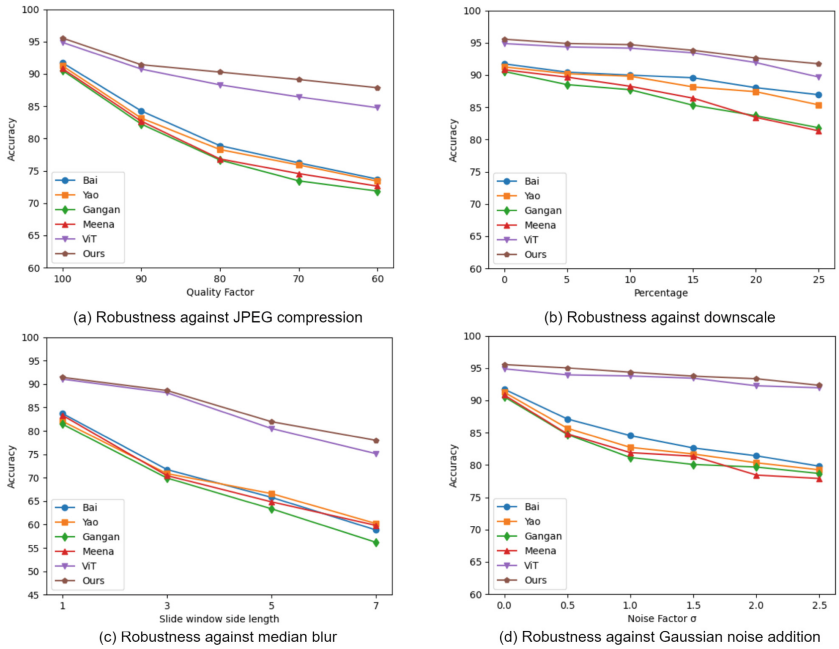

Fig. 4. The robustness against post-processing operations.

3.3 Evaluation of Generalization Ability

Cross-domain generalization poses a challenging problem for image forensics methods based on deep learning. Therefore, we demonstrate the performance of our method and other state-of-the-art methods in cross-domain scenarios in this section. We also adapt LSCGB dataset [10] as the benchmark training dataset and evaluate the detection performance on four separate datasets used for cross-domain testing: the Columbia dataset [25], consisting of 2400 images; the dataset proposed by Tokuda et al. [26], consisting of 4850 CG images and 4850 PG images; the dataset proposed by Rahmouni et al. [27], consisting of 3600 images; and the dataset proposed by He et al. [28], consisting of 6800 CG images and 6800 PG images.

As shown in Table 2, our model shows an improvement in accuracy compared to ViT across these four datasets. Moreover, compared to the approach proposed by Bai et al. [10], which performs the best among all CNN-based detection methods, our method respectively achieves 5.99%, 2.35%, 12.76%, and 6.53% improvements in cross-domain accuracy on the four datasets. These results highlight that our method also outperforms other CNN-based methods in cross-domain generalization.

Table 2. The detection accuracy on cross-domain datasets

Methods⟶ Datasets ↓	Bai [10]	Yao [11]	Gangan [12]	Meena [14]	ViT	Ours
Origin.	91.73	91.26	90.56	90.82	94.88	**95.55**
Columbia [25]	78.59	70.14	73.49	72.46	84.51	**84.58**
Tokuda [26]	91.70	90.91	87.45	85.28	93.95	**94.05**
Rahmouni [27]	76.83	75.39	69.42	73.14	87.75	**89.59**
He [28]	84.18	82.35	77.41	83.81	90.46	**90.71**

Table 3. Ablation experiments on the proposed method

Methods⟶ Scenarios ↓	ViT (Baseline)	w/o SRM Filter Block	w/o Conv. Block	Ours
Origin.	94.88	95.42	95.21	**95.55**
JPEG QF = 90	90.76	91.20	90.94	**91.43**
JPEG QF = 80	88.31	89.56	88.86	**90.29**
JPEG QF = 70	86.45	88.74	88.17	**89.13**
Rahmouni [27]	87.75	89.06	88.35	**89.59**

3.4 Ablation Study

In this section, we assess the convolutional block (Conv. Block) and the SRM filter block in terms of robustness against post-processing operations. We test their performances on the original dataset, the dataset edited by JPEG compression (quality factor (QF) $\in \{90, 80, 70\}$) and the dataset proposed by Rahmouni et al. [27].

As shown in Table 3, on the original dataset, the accuracy without the SRM filter block is 95.42%, and without the convolution block is 95.21%. Without the SRM filter block, the average accuracy declines by 0.45% under JPEG compression compared to our proposed method. Similarly, without the convolutional block, the average accuracy witnessed a decrease of 0.96% under the same conditions. On the dataset proposed by Rahmouni et al. [27], without the SRM filter block and the convolutional block, the accuracy decreases by 0.53% and 1.24%, respectively. The performance degradation confirms that utilizing the SRM filter block or the convolutional block effectively improves the performance of the model.

4 Conclusion

In this work, we propose a novel ViT with Forensic Feature Pre-processing (FFP) module for CG image forensics tasks. The advantage of ViT in capturing global features contributes to distinguishing CG images from PG images, and

the FFP module which exploits the discriminative information further improves the detection performance. Extensive experiments have shown that our method outperforms state-of-the-art methods, especially in terms of cross-domain generalization and robustness against post-processing operations. In further work, the proposed framework will also be extended and modified to tackle more image forensic applications, such as image tampering detection.

Acknowledgment. This work was supported in part by the National Natural Science Foundation of China (Nos. 62102100, 62102462), the Guangzhou Technology Plan Project (No. 202201011258), the Natural Science Foundation of Guang dong (Nos. 2022A1515 010108, 2023A1515011084), and the Talent fund of Guangdong Polytechnic Normal University (Nos. 2021SDKYA127, 2022SDKYA027, 99166990223).

References

1. Shum, H., Kang, S.B.: Review of image-based rendering techniques. In: Proceedings of SPIE, vol. 4067, pp. 2–13 (2000)
2. Goswami, P.: A survey of modeling, rendering and animation of clouds in computer graphics. Vis. Comput. **37**(7), 1931–1948 (2021)
3. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
4. Huang, H., li, Z., He, R., Sun, Z., Tan, T.: IntroVAE: introspective variational autoencoders for photographic image synthesis. In: Proceedings of 32nd International Conference on Neural Information Processing Systems, vol. 31, pp. 52–63 (2018)
5. Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., Bharath, A.A.: Generative adversarial networks: an overview. IEEE Signal Process. Mag. **35**(1), 53–65 (2018)
6. Goodfellow, I., et al.: Generative adversarial networks. Commun. ACM **63**(11), 139–144 (2020)
7. Nguyen, H.H., Tieu, T.N.D., Nguyen-Son, H.Q., Nozick, V., Yamagishi, J., Echizen, I.: Modular convolutional neural network for discriminating between computer-generated images and photographic images. In: Proceedings of the 13th International Conference on Availability, Reliability and Security, pp. 1–10 (2018)
8. Huang, R., Fang, F., Nguyen, H.H., Yamagishi, J., Echizen, I.: A method for identifying origin of digital images using a convolutional neural network. In: Proceedings of Asia-Pacific Signal Information Processing Association Annual Summit and Conference (APSIPA ASC), pp. 1293–1299 (2020)
9. Zhang, R.S., Quan, W.Z., Fan, L.B., Hu, L.M., Yan, D.M.: Distinguishing computer-generated images from natural images using channel and pixel correlation. J. Comput. Sci. Technol. **35**, 592–602 (2020)
10. Bai, W., et al.: Robust texture-aware computer-generated image forensic: benchmark and algorithm. IEEE Trans. Image Process. **30**, 8439–8453 (2021)
11. Yao, Y., Zhang, Z., Ni, X., Shen, Z., Chen, L., Xu, D.: CGNet: detecting computergenerated images based on transfer learning with attention module. Signal Process. Image Commun. **105**, 116692 (2022)
12. Gangan, M.P., Anoop, K., Lajish, V.: Distinguishing natural and computer generated images using Multi-Colorspace fused EfficientNet. J. Inf. Secur. Appl. **68**, 103261 (2022)

13. Tan, M., Le, Q.: EfficientNet: rethinking model scaling for convolutional neural networks. In: Proceedings of International Conference on Machine Learning, pp. 6105–6114 (2019)
14. Meena, K.B., Tyagi, V.: Distinguishing computer-generated images from photographic images using two-stream convolutional neural network. Appl. Soft Comput. **100**, 107025 (2021)
15. Fridrich, J., Kodovsky, J.: Rich models for steganalysis of digital images. IEEE Trans. Inf. Forensics Secur. **7**(3), 868–882 (2012)
16. Dosovitskiy, A., et al.: An image is worth 16×16 words: transformers for image recognition at scale. In: Proceedings of International Conference on Learning Representations (2020)
17. Sun, Y., Ni, R., Zhao, Y.: ET: edge-enhanced transformer for image splicing detection. IEEE Signal Process. Lett. **29**, 1232–1236 (2022)
18. Heo, Y.J., Yeo, W.H., Kim, B.G.: Deepfake detection algorithm based on improved vision transformer. Appl. Intell. **53**(7), 7512–7527 (2023)
19. Li, G., Yao, H., Le, Y., Qin, C.: Recaptured screen image identification based on vision transformer. J. Vis. Commun. Image Represent. **90**, 103692 (2023)
20. Yuan, K., Guo, S., Liu, Z., Zhou, A., Yu, F., Wu, W.: Incorporating convolution designs into visual transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 579–588 (2021)
21. Mao, X., et al.: Towards robust vision transformer. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12042–12051 (2022)
22. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv preprint arXiv:1607.06450 (2016)
23. Hendrycks, D., Gimpel, K.: Gaussian error linear units (GELUs). arXiv preprint arXiv:1606.08415 (2016)
24. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
25. Ng, T.T., Chang, S.F., Hsu, J., Pepeljugoski, M.: Columbia photographic images and photorealistic computer graphics dataset. Columbia University, ADVENT Technical Report, pp. 205–2004 (2005)
26. Tokuda, E., Pedrini, H., Rocha, A.: Computer generated images vs. digital photographs: a synergetic feature and classifier combination approach. J. Vis. Commun. Image Represent. **24**(8), 1276–1292 (2013)
27. Rahmouni, N., Nozick, V., Yamagishi, J., Echizen, I.: Distinguishing computer graphics from natural images using convolution neural networks. In: 2017 IEEE Workshop on Information Forensics and Security (WIFS), pp. 1–6. IEEE (2017)
28. He, P., Jiang, X., Sun, T., Li, H.: Computer graphics identification combining convolutional and recurrent neural networks. IEEE Signal Process. Lett. **25**(9), 1369–1373 (2018)