Computer Fraud and Security
ISSN (online): 1873-7056

IFDMamba: An Image Forgery Detection Method Based on Context-aware Mamba

Aifan Zhuang[1], Zhijun Chen[1,*], Xiaozhao Li[2], Yimiao Liu[3,4], Zhiguang Lv[5], Yajing Shi[6]

[1]*School of Law, People's Public Security University of China, Beijing 100038, China*

[2]*Institute of Electrical Engineering and Advanced Electromagnetic Drive Technology, Qilu Zhongke, Jinan 250013, Shandong, China*

[3]*Institute of Electrical Engineering, Chinese Academy of Sciences, Beijing 100190, China*

[4]*School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing 100049, China*

[5]*Faculty of Electrical and Control Engineering, Liaoning Technical University, Huludao 125105, Liaoning, China*

[6]*Institute of Computing Technology, China Academy of Railway Science Corporation Limited, Beijing 100081, China*

**Corresponding Author.*

Abstract:

To address the challenges in existing image forgery detection methods, including the difficulty of effectively capturing and integrating local fine-grained features with global background features and the overlooked relative definition of pristine and forged pixels within a single image, we propose IFDMamba, a context-aware Mamba-based Image Forgery Detection method. Firstly, we propose a novel context-aware Mamba, which enhances local contextual relationships between image patches by constructing a Gated Spatial Convolution (GaSC) module. Additionally, a bidirectional Mamba model is introduced to capture the global contextual relationships across the entire sequence of image patches. This enables the effective extraction and complementary integration of local fine-grained features and global background features in forged images, facilitating accurate localization of forged regions in complex backgrounds. Secondly, we propose an improved NT-Xent contrastive loss tailored for image forgery detection tasks, utilizing pixel-wise contrastive learning to supervise the extraction of high-level forensic features for each image. This loss function effectively captures the inherent distinction between pristine and forged pixels within an image. Finally, during the model testing phase, we use K-means to map the extracted high-level forensic features to the predicted forgery masks in real-time, further minimizing cross-image interference in the training data. Experimental results demonstrate that IFDMamba achieves consistent performance improvements over mainstream methods on five public datasets—Coverage, NIST, CASIA, MISD, and FF++. The method exhibits strong forgery detection capability and robustness in complex backgrounds, and holds significant application value in combating criminal networks in the black and gray markets that rely on image forgery.

Keywords: image evidence, image forgery detection, Mamba, context-aware, contrastive learning

INTRODUCTION

As image editing tools and generative models continue to evolve, ordinary users can now easily manipulate digital images without specialized knowledge. The convenience and high realism of image forgery techniques present significant challenges to legislative, judicial, and administrative oversight in combating criminal activities in the black and gray markets. The rise of new types of internet-based crimes relying on image forgery is exhibiting a trend of industrialization and sophistication, severely threatening citizens' rights and social security. Meanwhile, the technical barriers to digital forensics hinder the collection, traceability, and detection of electronic evidence. To effectively address these challenges, innovating image forgery detection technologies is a logical and necessary approach to tackling the key pain points in managing criminal networks in the black and gray markets.

Traditional image forgery detection methods [1-4] mainly rely on handcrafted features to identify forged regions in images. However, these methods struggle to generalize when faced with diverse and complex forgery techniques, exhibiting significant limitations. In recent years, classification-based deep learning methods [5-8] have significantly outperformed traditional approaches in terms of performance. These methods automatically learn forensic features from images, enabling more accurate detection and localization of forged regions, and demonstrating stronger generalization ability when dealing with unseen forgeries. Some studies focus on detecting specific types of forgeries, such as splicing [9], copy-move [10], and inpainting [11], achieving good results in those particular tasks. Additionally, more robust and practical solutions have been developed to detect complex and mixed forgeries, even when they are accompanied by transmission degradation and various post-processing operations [5-7,12].

Although existing image forgery detection methods have achieved notable progress, they still face several challenges. First, many approaches focus on extracting features from local regions of the image, such as forgery boundary artifacts [5,6], compression artifacts [12], and noise features [7]. However, these methods often overlook the integration of global information when handling complex backgrounds. The lack of effective complementarity between local fine-grained features and global background features makes it difficult for detection models to accurately identify forgery regions, especially in the presence of complex backgrounds or mixed types of forgeries. Second, existing deep learning-based detection methods typically simplify the problem as a binary classification task, categorizing pixels as either pristine or forged. However, the definition of pristine and forged pixels is only relative within a single image, a nuance that is often ignored by these methods. As a result, forged (or pristine) regions from different images are unnecessarily mixed into the same category, leading to potential misclassifications.

In recent years, State Space Models (SSMs) have been introduced into the field of deep learning [13-15]. Inspired by continuous state space models in control systems and combined with HiPPO [16] initialization, the Linear State Space Layer (LSSL) [14] has shown effectiveness in handling long-range dependencies. However, LSSL has high computational and memory requirements, which makes it difficult to apply to complex tasks. To address this issue, the Structured State-Space Sequence Model (S4) [13] was proposed, which normalizes parameters into a diagonal structure and has become a potential alternative to CNNs or Transformers. Since then, various structured SSMs have been proposed, including a data-dependent SSM introduced by Gu and Dao [17], which builds a universal language model backbone called Mamba. Mamba surpasses Transformers in handling large-scale real-world datasets, showcasing its ability to scale linearly with increasing data size. Following the success of S4 in long-sequence data modeling, researchers have gradually extended their work to the visual domain. S4ND [18], the first model to apply SSMs to visual tasks, demonstrated its potential to compete with ViT [19]. DiffuSSM [20] further employs an SSM-based backbone to replace traditional attention mechanisms, enabling the generation of high-resolution images at an acceptable computational cost.

To address the challenges in image forgery detection, and inspired by the ability of SSMs to handle long-range dependencies and global contextual relationships in visual tasks, we propose IFDMamba, an image forgery detection method based on context-aware Mamba. This method extracts both local fine-grained features and global background features from forged images using context-aware Mamba. It also employs pixel-wise contrastive learning to explicitly model the relative nature of pristine and forged pixels within a single image. Our primary contributions are outlined as follows:

(1) We propose a novel context-aware Mamba model, which effectively enhances local contextual relationships between image patches using a newly introduced Gated Spatial Convolution (GaSC) module. Additionally, by incorporating a bidirectional Mamba modeling mechanism, the model captures global contextual relationships across the entire sequence of image patches. This enables the effective extraction and complementary integration of local fine-grained features and global background features in forged images, enhancing the model's precision in localizing forged regions within complex backgrounds.

(2) We design an improved NT-Xent contrastive loss tailored for image forgery detection tasks. This loss function employs pixel-wise contrastive learning to supervise the extraction of high-level forensic features on a per-image basis, explicitly modeling the relative relationship between pristine and forged pixels within a single image. This approach effectively improves both the precision and robustness of forgery detection.

(3) Extensive experiments were conducted on five public datasets—Coverage, NIST, CASIA, MISD, and FF++. The results demonstrate that IFDMamba accurately detects forged regions in complex backgrounds and significantly reduces false positives, outperforming existing mainstream methods.

METHODOLOGY

This section elaborates on the conceptual framework and practical implementation details of IFDMamba. First, we introduce the foundational concepts related to SSMs, including both their continuous and discrete forms, and provide an overview of the efficient computational methods for selective SSM (Mamba). Next, we present the overall framework of the model and conduct an in-depth analysis of its key components, including the context-aware mamba encoder and the GaSC module. Following this, we introduce the improved NT-Xent contrastive loss and discuss the model testing approach based on unsupervised clustering. Finally, we analyze the algorithmic complexity of IFDMamba.

Preliminaries

State space models

SSMs are conceptualized as linear time-invariant (LTI) systems that transform an input sequence $x(t) \in \mathbb{R}^L$ into an output sequence $y(t) \in \mathbb{R}^L$ via a hidden state. SSMs are generally described using the following system of linear ordinary differential equations (ODEs):

$$\begin{aligned} h'(t) &= \boldsymbol{A}h(t) + \boldsymbol{B}x(t) \\ y(t) &= \boldsymbol{C}h(t) + Dx(t) \end{aligned} \tag{1}$$

where $\boldsymbol{A} \in \mathbb{C}^{N\times N}$, $\boldsymbol{B}, \boldsymbol{C} \in \mathbb{C}^N$, and $D \in \mathbb{C}^1$ are the weight parameters.

Discretization of SSMs

To integrate continuous-time SSMs into deep neural network models, discretization is required. This process can be achieved by solving the ODE and applying a straightforward discretization method. Specifically, the analytical solution to Eq. (1) can be expressed as:

$$h(t_n) = e^{\boldsymbol{A}(t_m - t_n)}h(t_m) + \qquad e^{\boldsymbol{A}(t_m - t_n)} \int_{t_m}^{t_n} \boldsymbol{B}(\gamma)x(\gamma)e^{-\boldsymbol{A}(\gamma - t_m)}d\gamma \tag{2}$$

Next, by sampling with a step size $\boldsymbol{\Delta}$ (i.e., $d\gamma \Big|_{t_i}^{t_{i+1}} = \Delta_i$), we can discretize $h(t_n)$ as:

$$h_n = e^{\boldsymbol{A}(\Delta_m + \cdots + \Delta_{n-1})}(h_m + \textstyle\sum_{i=m}^{n-1} \boldsymbol{B}_i\, x_i e^{-\boldsymbol{A}(\Delta_m + \cdots + \Delta_i)}\Delta_i) \tag{3}$$

This discretization is approximately equivalent to the result obtained using the zero-order hold (ZOH) method, which is commonly used in SSM-related literature. Specifically, let $n = m + 1$, then Eq. (3) can be written as:

$$\begin{aligned} h_{m+1} &= e^{\boldsymbol{A}\Delta_m}(h_m + \boldsymbol{B}_m x_m e^{-\boldsymbol{A}\Delta_m}\Delta_m) \\ &= e^{\boldsymbol{A}\Delta_m}h_m + \boldsymbol{B}_m\Delta_m x_m \\ &= \overline{\boldsymbol{A}_m}h_m + \overline{\boldsymbol{B}_m}x_m \end{aligned} \tag{4}$$

Here, $\overline{\boldsymbol{A}_m} = e^{\boldsymbol{A}\Delta_m}$ aligns with the discretization result of ZOH, while $\overline{\boldsymbol{B}_m} = \boldsymbol{B}_m\Delta_m$ is essentially the first-order Taylor expansion of the corresponding result obtained through ZOH.

Selective SSMs

To overcome the limitations of linear time-invariant SSMs in capturing contextual information, the weight matrices B, C, the weight parameter D, and the step size $\boldsymbol{\Delta}$ in Eq. (1) are configured as input-dependent functions [17]. However, the resulting time-varying SSMs introduce challenges in efficient computation, as convolution operations do not support dynamic weights. Nevertheless, if the recurrence relation for h_n in Eq. (3) can be derived, it can still be computed efficiently. Specifically, let $e^{\boldsymbol{A}(\Delta_m + \cdots + \Delta_{i-1})}$ be denoted as $\boldsymbol{u}_{\boldsymbol{A},m}^i$, and its recurrence relation can be written as:

$$\boldsymbol{u}_{\boldsymbol{A},m}^i = e^{\boldsymbol{A}\Delta_{i-1}}\boldsymbol{u}_{\boldsymbol{A},m}^{i-1} \tag{5}$$

For the second term in Eq. (3), we obtain:

$$\begin{aligned} \boldsymbol{u}_{\boldsymbol{B},m}^n &= e^{\boldsymbol{A}(\Delta_m + \cdots + \Delta_{n-1})} \textstyle\sum_{i=m}^{n-1} \boldsymbol{B}_i\, x_i e^{-\boldsymbol{A}(\Delta_m + \cdots + \Delta_i)}\Delta_i \\ &= e^{\boldsymbol{A}\Delta_{n-1}}\boldsymbol{u}_{\boldsymbol{B},m}^{n-1} + \boldsymbol{B}_{n-1}x_{n-1}\Delta_{n-1} \end{aligned} \tag{6}$$

Thus, using Eq. (5) and (6), we can derive:

$$h_n = \boldsymbol{u}_{\boldsymbol{A},m}^n h_m + \boldsymbol{u}_{\boldsymbol{B},m}^n \tag{7}$$

This expression can be efficiently computed in parallel using associative scan algorithms [15,21], which effectively reduce the overall computational complexity to linear. Furthermore, IFDMamba accelerates the computation by utilizing hardware-aware algorithms [17].

Model Architecture

As illustrated in Figure 1, the IFDMamba framework comprises two stages: the training phase and the testing phase. In the training phase, the model first extracts high-level forensic features F from the given input image X using the context-aware Mamba. The context-aware Mamba enhances local contextual relationships between image patches by constructing the Gated Spatial Convolution (GaSC) module. It also introduces a bidirectional Mamba model to capture the global contextual relationships across the entire sequence of image patches, enabling effective extraction and complementary integration of both local fine-grained features and global background features from the input image. Next, the improved NT-Xent contrastive loss is applied to perform pixel-wise contrastive learning on the high-level forensic features F for each individual image. The ground truth forgery mask Y naturally provides positive and negative class labels, making the pixel-wise contrastive learning more effective. Moreover, this per-image training approach distinguishes itself from existing methods [8,9,22], which train on entire mini-batches. It better captures the relative nature of pristine and forged pixels within a single image.

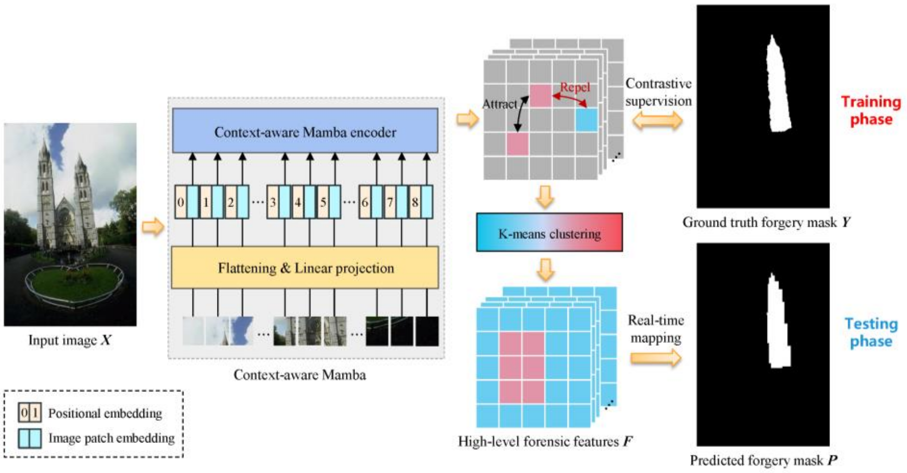


Figure 1. The Overall Framework of IFDMamba

In the testing phase, the test image is passed through the trained context-aware Mamba to generate high-level forensic features F, which are then mapped in real-time to the predicted forgery mask P using the K-means clustering algorithm. Assuming that the forged pixels occupy a relatively minor proportion of the overall image, the cluster label containing the majority of elements is assigned to pristine pixels, while the remaining labels are designated as forged pixels. This testing framework not only resolves the feature mapping issue but also reduces the cross-image interference present in the training data, enabling IFDMamba to more accurately detect forged regions when processing new images.

Context-Aware Mamba

The standard Mamba model is designed for one-dimensional sequences. As depicted in Figure 1, to address the image forgery detection task, the input image $\boldsymbol{X} \in \mathbb{R}^{H\times W\times C}$ is first transformed into flattened 2-D image patches $\boldsymbol{x}_J \in \mathbb{R}^{E\times(J^2\cdot C)}$. Here, (H, W) denote the height and width of the input image, C represents the number of channels, J specifies the patch size, and E indicates the total number of image patches. Subsequently, $\boldsymbol{x}_J$ undergoes a linear projection to form a vector of size D, and positional embeddings $\boldsymbol{E}_{pos} \in \mathbb{R}^{E\times D}$ are added as follows:

$$\boldsymbol{X}_0 = \left[\boldsymbol{x}_J^1\boldsymbol{W}; \boldsymbol{x}_J^2\boldsymbol{W};; \boldsymbol{x}_J^E\boldsymbol{W}\right] + \boldsymbol{E}_{pos} \tag{8}$$

where $\boldsymbol{x}_J^e$ denotes the e -th patch of image X, while $\boldsymbol{W} \in \mathbb{R}^{(J^2\cdot C)\times D}$ is a trainable projection matrix.

The sequence of image patches $\boldsymbol{X}_{l-1}$ is subsequently fed into the l-th layer of the context-aware Mamba encoder, yielding the output $\boldsymbol{X}_l$. Finally, the output $\boldsymbol{X}_L$ is normalized to obtain the final high-level forensic feature $\boldsymbol{F} \in \mathbb{R}^{\widehat{H}\times\widehat{W}\times\hat{C}}$, where $\widehat{H} = \left(\frac{H-J}{S}\right) + 1$, $\widehat{W} = \left(\frac{W-J}{S}\right) + 1$, and S is the stride used to extract image patches, while $\hat{C}$ is the dimensionality of the output embedding space. The above process is summarized as:

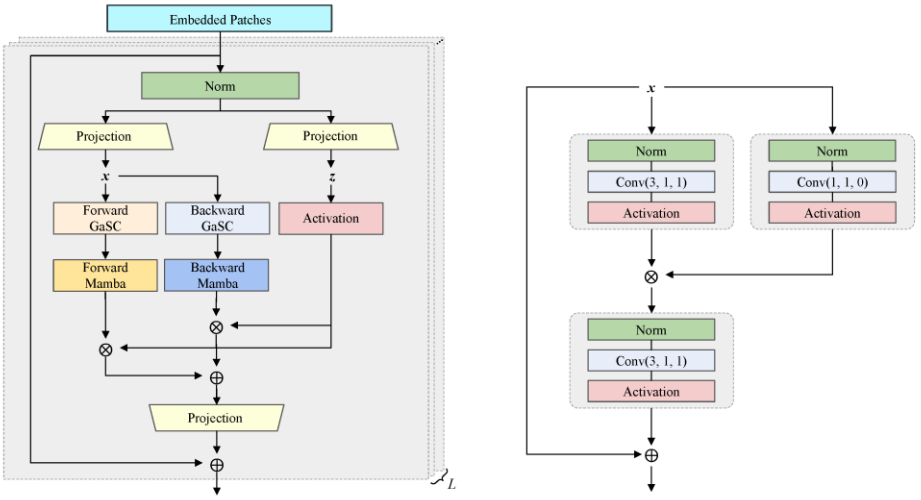
$$\boldsymbol{X}_l = Encoder(\boldsymbol{X}_{l-1}) + \boldsymbol{X}_{l-1} \tag{9}$$

$$\boldsymbol{F} = Norm(\boldsymbol{X}_L) \tag{10}$$

where the Encoder denotes the proposed context-aware Mamba encoder, L denotes the total number of layers, while Norm refers to the normalization layer.

Context-aware mamba encoder

The standard Mamba block is optimized for processing one-dimensional sequences and is not suitable for vision tasks that require spatial awareness. In this section, we introduce the context-aware Mamba encoder, which integrates the GaSC module and a bidirectional modeling mechanism to adapt to image forgery detection tasks. Its structure is shown in Figure 2(a).



(a) Context-aware Mamba encoder

(b) GaSC

Figure 2. Architecture of the context-aware mamba encoder and the GaSC module

Table 1. Context-aware mamba encoder

```
Algorithm 1: Context-Aware Mamba Encoder
Input: Image patch sequence $\boldsymbol{X}_{l-1}$
Output: Image patch sequence $\boldsymbol{X}_l$
// Normalize the input image patch sequence
    $\boldsymbol{X}'_{l-1} \leftarrow Norm(\boldsymbol{X}_{l-1})$
    $\boldsymbol{x} \leftarrow Linear^{x}(\boldsymbol{X}'_{l-1})$
    $\boldsymbol{z} \leftarrow Linear^{z}(\boldsymbol{X}'_{l-1})$
// Bidirectional processing
    for $v$ in {forward, backward} do
        $\boldsymbol{X}'_v \leftarrow SiLU(GaSC_v(\boldsymbol{x}))$
        $\boldsymbol{B}_v \leftarrow Linear^{B}_v(\boldsymbol{X}'_v)$
        $\boldsymbol{C}_v \leftarrow Linear^{C}_v(\boldsymbol{X}'_v)$
        $\Delta_v \leftarrow \log(1 + \exp(Linear^{\Delta}_v(\boldsymbol{X}'_v) + Parameter^{\Delta}_v))$
        $\overline{\boldsymbol{A}_v} \leftarrow \Delta_v \otimes Parameter^{A}_v$
        $\overline{\boldsymbol{B}_v} \leftarrow \Delta_v \otimes \boldsymbol{B}_v$
        $\boldsymbol{y}_v \leftarrow SSM(\overline{\boldsymbol{A}_v}, \overline{\boldsymbol{B}_v}, \boldsymbol{C}_v)(\boldsymbol{X}'_v)$
    end for
    $\boldsymbol{y}'_{\text{forward}} \leftarrow \boldsymbol{y}_{\text{forward}} \odot SiLU(\boldsymbol{z})$
```

```
$\boldsymbol{y}'_{\text{backward}} \leftarrow \boldsymbol{y}_{\text{backward}} \odot SiLU(\boldsymbol{z})$
// Residual connections
$\boldsymbol{X}_l \leftarrow Linear^{X}(\boldsymbol{y}'_{\text{forward}} + \boldsymbol{y}'_{\text{backward}}) + \boldsymbol{X}_{l-1}$
return: $\boldsymbol{X}_l$
```

Specifically, the operations of the context-aware Mamba encoder are described in detail in Table 1. The input image patch sequence $\boldsymbol{X}_{l-1}$ is first normalized. Then, the normalized sequence is linearly projected into two components, x and z. Next, the x undergoes processing in both forward and backward directions. This bidirectional modeling mechanism captures the global contextual relationships within the image patch sequence, effectively enhancing the extraction of global background features.

For each direction, x is first passed through the GaSC module to obtain $\boldsymbol{x}'_v$. The $\boldsymbol{x}'_v$ is then linearly projected into $\boldsymbol{B}_v$, $\boldsymbol{C}_v$, and $\boldsymbol{\Delta}_v$, and $\boldsymbol{\Delta}_v$ is used to transform $\overline{\boldsymbol{A}_v}$ and$\overline{\boldsymbol{B}_v}$. The Mamba model is then applied to compute $\boldsymbol{y}_{\text{forward}}$ and $\boldsymbol{y}_{\text{backward}}$. The outputs $\boldsymbol{y}_{\text{forward}}$ and $\boldsymbol{y}_{\text{backward}}$ are gated by $SiLU(\boldsymbol{z})$, resulting in $\boldsymbol{y}'_{\text{forward}}$ and $\boldsymbol{y}'_{\text{backward}}$. Finally, the gated outputs undergo a linear transformation and are summed before being combined with the residual connection from $\boldsymbol{X}_{l-1}$, resulting in the output sequence $\boldsymbol{X}_l$.

Gated Spatial Convolution (GaSC)

The context-aware Mamba models feature dependencies by flattening the 2D image into a 1D sequence. To effectively extract local fine-grained features and enhance the local context relationships between image patches before the bidirectional Mamba modeling, a GaSC module is designed. As illustrated in Figure 2(b), the input image patch sequence x is fed in parallel into two convolution blocks to capture and represent features across various hierarchical levels. In the feature extraction block, the input x is normalized through layer normalization and then passed into a convolutional layer configured with a kernel size of $k = 3$, stride $s = 1$, and padding $p = 1$. The output then passes through a ReLU activation function. This convolution block applies a small convolution kernel to adjacent image patches, enabling the extraction of subtle local features. In the gating generation block, the input x undergoes normalization, followed by processing through a convolutional layer configured with a kernel size of $k = 1$, stride $s = 1$, and padding $p = 0$. Subsequently, a Sigmoid activation function is applied to the resulting output.

The outputs of the feature extraction block and gating generation block are then multiplied pixel-by-pixel. This gating mechanism dynamically weights the feature channels using the gating vector, emphasizing important feature information and suppressing irrelevant or redundant features, effectively controlling the flow of information. Finally, the features are further fused using the feature extraction block, with a residual connection reusing the input features. The process can be represented as:

$$\text{GaSC}(\boldsymbol{x}) = \boldsymbol{x} + \text{Conv}^{3,1,1}(\text{Conv}^{3,1,1}(\boldsymbol{x}) \cdot \text{Conv}^{1,1,0}(\boldsymbol{x})) \tag{11}$$

where $\text{Conv}^{k,s,p}$ denotes a convolutional block characterized by a kernel size of k, a stride of s, and a padding of p. The convolutional layers in Figure 2(b) follow the same notation.

Improved NT-Xent Contrastive Loss

To preserve the relative relationship between original and forged pixels within a single image, we propose an improved NT-Xent contrastive loss for pixel-wise contrastive learning in IFDMamba. Specifically, we first process the high-level forensic features F by applying a flattening operation $f(\cdot): \mathbb{R}^{\hat{H}\times\hat{W}\times\hat{C}} \to \mathbb{R}^{\hat{H}\hat{W}\times\hat{C}}$, as follows:

$$f(\boldsymbol{F}) \to \{\boldsymbol{q}, \boldsymbol{k}_1^+, \boldsymbol{k}_2^+, \cdots \boldsymbol{k}_K^+, \boldsymbol{k}_1^-, \boldsymbol{k}_2^-, \cdots \boldsymbol{k}_P^-\} \tag{12}$$

Here, we define the dictionary as $\{\boldsymbol{q}, \boldsymbol{k}_1^+, \boldsymbol{k}_2^+, \cdots \boldsymbol{k}_K^+, \boldsymbol{k}_1^-, \boldsymbol{k}_2^-, \cdots \boldsymbol{k}_P^-\}$, where q is the encoded query vector. The set $\{\boldsymbol{q}, \boldsymbol{k}_1^+, \boldsymbol{k}_2^+, \cdots \boldsymbol{k}_K^+\}$ represents features extracted from pristine regions, indexed as 0 in the ground truth forgery mask $\boldsymbol{Y} \in \{0,1\}^{\hat{H}\times\hat{W}}$. Similarly, $\{\boldsymbol{k}_1^-, \boldsymbol{k}_2^-, \cdots \boldsymbol{k}_P^-\}$ corresponds to features from forged regions, indexed as 1 in Y. In image forgery detection tasks, both pristine and forged regions often span multiple pixels, resulting in the dictionary containing a significantly larger number of positive keys ($K > 1$). Thus, the improved NT-Xent contrastive loss function tailored for image forgery detection can be expressed as:

$$\mathcal{L}_{\text{NT-Xent+}} = -log\frac{\frac{1}{K}\sum_{i=1}^{K}\exp\left(sim(\boldsymbol{q},\boldsymbol{k}_i^{+})/\tau\right)}{\sum_{j=1}^{P}\exp\left(sim(\boldsymbol{q},\boldsymbol{k}_j^{-})/\tau\right)} \tag{13}$$

where τ is the temperature hyperparameter [23], and $sim(\cdot,\cdot)$ denotes the cosine similarity. In the original NT-Xent loss [24], each query vector q only matches one positive key in the dictionary. However, in the improved NT-Xent loss (Eq. (13)), all positive keys are considered in the loss calculation, which is accomplished by calculating the expected value of the dot product between q and the elements in the set $\{\boldsymbol{k}_i^{+}\}$.

As depicted in Figure 1, supervision during the training phase involves directly aligning the ground truth forgery mask Y with the extracted high-level forensic features F, without generating a predicted forgery mask $P \in \mathbb{R}^{\hat{H}\times\hat{W}}$. Additionally, for every image within the minibatch, the contrastive loss $\mathcal{L}_{\text{NT-Xent+}}$ is computed on an image-by-image basis and subsequently aggregated to calculate the total loss, rather than being computed over the entire batch. Specifically, given a mini-batch of features $\{\boldsymbol{F}_1, \boldsymbol{F}_2 \cdots, \boldsymbol{F}_B\}$, the total contrastive loss $\mathcal{L}$ is computed as follows:

$$\mathcal{L} = \frac{1}{B}\sum_{i=1}^{B}\mathcal{L}_{\text{NT-Xent+}}(\boldsymbol{F}_i) \tag{14}$$

In Eq. (14), the mini-batch features are not merged to compute the overall $\mathcal{L}_{\text{NT-Xent+}}$, which helps avoid cross-image influences within the training data. This loss function is designed based on the relative definition of original and forged pixels within a single image, which significantly differs from the batch-level loss computation used in methods like [8,24,25].

It is worth noting that, compared to the original NT-Xent loss function, the improved NT-Xent contrastive loss may increase the computation time. The main reason is that the query vector q now matches with K positive keys instead of just one, leading to an increased number of positive similarity calculations. Moreover, the loss function is computed on an image-by-image basis (as shown in Eq. (14)), which may further increase the computational overhead. However, as demonstrated in Section 3.4.2, the additional computational cost is within an acceptable range, and a good balance between model performance and time efficiency is achieved.

Model Testing with Unsupervised Clustering

This section outlines the testing phase of IFDMamba in detail. A major challenge during the testing phase is effectively mapping the extracted high-level forensic features to a predicted forgery mask. In contrast to conventional approaches that rely on pre-trained classifiers, we adopt an unsupervised, online learning method. As previously mentioned, the classification of pristine and forged pixels is relative within a single image and does not generalize well across different images. This highlights the limitations of earlier classification-based detection methods, where classifiers trained on the training datasets often fail to generalize to previously unseen test samples. Therefore, mapping the features of each image separately to the final forgery mask is a more reasonable solution.

Specifically, we use K-means to cluster the high-level forensic features F. The cluster containing the majority of elements is labeled as pristine while the others is labeled as forged under the assumption that forged pixels constitute only a small fraction of the overall image. The high-level forensic features F extracted through context-aware Mamba and pixel-wise contrastive learning enable effective extraction and complementary integration of local fine-grained features and global background features from the test image. These features exhibit strong discriminative power, making unsupervised clustering sufficient for the task.

Algorithm Complexity Analysis

The computational complexity of IFDMamba can be assessed by examining its three primary components: linear projection, context-aware Mamba encoder, and the improved NT-Xent contrastive loss. First, in the linear projection phase, the complexity of computing the linear projection for each image patch is $\mathrm{O}(\mathrm{J}^2\mathrm{CD})$. Here, J represents the patch size, C denotes the number of channels, and D indicates the feature dimension. For E image patches, the total complexity for the linear projection is $O(EJ^2CD)$. Second, for the context-aware Mamba encoder, the computational complexity is primarily influenced by the GaSC module and the Mamba module. The complexity of the GaSC module primarily arises from the convolution operations. When performing convolution on a 1-D sequence, the complexity depends on the number of image patches E, the input and output channel dimensions D_{in} and D_{out}, and the kernel size k. When the input/output channels are large and k is constant, the complexity can be simplified to $O(ED_{in}D_{out})$. The Mamba module plays a crucial role in adaptively capturing global context, akin to the self-attention mechanism used in Transformers. The complexity of the Mamba module is also $O(ED_{in}D_{out})$, which is significantly lower than the quadratic complexity of self-attention, thus reducing computational resource consumption while still capturing global contextual relationships effectively. Additionally, as described in Section 2.1, the derivation of Mamba's expressions can be efficiently computed using the associated scanning algorithm [15,21], and further accelerated by hardware-aware algorithms [17]. Finally, for the improved NT-Xent contrastive loss, each query vector q needs to compute similarity with K positive samples and P negative samples. Given the input high-level forensic features $F \in \mathbb{R}^{\widehat{H} \times \widehat{W} \times \hat{C}}$, for a mini-batch size of B, the computational complexity is $O(B\widehat{H}\widehat{W}(K+P)\hat{C})$.

In summary, the computational complexity of IFDMamba scales linearly with the number of image patches E (i.e., image size). This indicates that IFDMamba is highly efficient in processing high-resolution images, allowing it to control computational resource consumption while maintaining strong model performance, thus meeting the requirements of practical applications. Additionally, by incorporating model compression and pruning techniques, the number of model parameters and the computational load can be further reduced. Furthermore, training strategies such as mixed-precision training and gradient checkpointing can be employed to lower memory usage and computational overhead during the training process.

EXPERIMENT

To comprehensively evaluate the proposed IFDMamba, we carried out a series of experiments focused on answering the following research questions:

Q1: How does IFDMamba perform quantitatively and qualitatively compared to mainstream image forgery detection methods?

Q2: Do the core components of IFDMamba, including the bidirectional modeling mechanism of the context-aware Mamba encoder, the GaSC module, and the improved NT-Xent contrastive loss, play a critical role in enhancing its detection performance?

Experimental Setup

Datasets

Training Datasets IFDMamba is trained using the same datasets as described in [8,12], specifically the tamperedCOCO and tamperedRAISE datasets.

TamperedCOCO: This dataset comprises two subsets, SP COCO and CM COCO, both derived from the COCO dataset. SP COCO is designed specifically for splicing forgery and contains approximately 200,000 forged images. CM COCO is tailored for copy-move forgery, also comprising around 200,000 forged images.

TamperedRAISE: This dataset includes three subsets: CM RAISE, CM-JPEG RAISE, and JPEG RAISE, all created from the RAISE dataset. It contains approximately 400,000 forged images in total. CM RAISE is designed for copy-move forgery, CM-JPEG RAISE is derived from CM RAISE with additional image compression applied, and JPEG RAISE is created by compressing images directly from the original RAISE dataset.

Testing Datasets We evaluated IFDMamba using five commonly used datasets: Coverage [26], NIST [27], CASIA [28], MISD [29], and FF++ [30]. These datasets encompass a wide variety of highly complex forged images, providing a comprehensive testbed for forgery detection algorithms.

Coverage: This dataset includes forged images specifically designed to challenge similarity-based copy-move forgery detectors. It intentionally introduces similar but authentic objects to increase detection difficulty.

NIST: This dataset contains a wide range of forged images, including splicing, copy-move, software manipulations, and post-processing operations. It is characterized by high-resolution images.

CASIA: A commonly used dataset for detecting copy-move and splicing forgeries, with diverse image sources. Since ground truth forgery masks are not officially provided, masks created by a third-party user [31] were used.

MISD: This dataset includes multi-source forged images, where forged images are generated by combining content from multiple distinct sources.

FF++: A dataset containing facial images synthesized using Generative Adversarial Networks (GANs).

Importantly, the training and testing datasets are completely distinct, ensuring that the evaluation reflects real-world conditions and effectively measures the ability of forgery detection algorithms to generalize.

Baselines and evaluation metrics

We selected the following five mainstream baseline models for comparison:

PSCC-Net [6]: An end-to-end image forgery detection model that employs a high-resolution feature network as the backbone and incorporates a progressive spatiotemporal attention mechanism to capture contextual information.

MVSS-Net [5]: This model leverages multi-view feature learning and multi-scale supervision to detect and identify forged regions by exploiting noise distribution features and boundary artifacts around the forged areas.

IF-OSN [7]: A model designed to capture predictable and imperceptible noise introduced by online social platforms.

CAT-Net [12]: An end-to-end fully convolutional neural network that jointly learns RGB image features and compressed forgery features in the DCT domain.

TruFor [8]: A Transformer-based fusion architecture that extracts high- and low-level scale features by combining RGB images with noise-sensitive fingerprints, enabling the detection of diverse local forgeries.

To ensure a fair comparison, PSCC-Net, MVSS-Net, and IF-OSN were re-trained on identical training datasets.

Following the convention [5,8,12], two commonly used metrics are employed to evaluate the performance of IFDMamba: the F1 score and Intersection over Union (IoU). The macro-averaged F1 score is formally defined as follows:

$$F1 = \frac{1}{X}\sum_{x=1}^{X}\frac{2\times TP_x}{2\times TP_x+FP_x+TN_x} \tag{15}$$

The IoU metric is computed as:

$$\text{IoU} = \frac{\boldsymbol{P\cap Y}}{\boldsymbol{P\cup Y}} \tag{16}$$

where P and Y represent the predicted forgery mask and the ground truth forgery mask, respectively.

Experimental details

IFDMamba is implemented using the PyTorch framework, with the Adam optimizer (using default parameters) and an initial learning rate of $1e-4$. The training batch size is set to 4, and all input images are resized to 1024×1024. A projection layer with a convolution kernel size of 16×16 is employed to obtain non-overlapping image patch sequences. The context-aware Mamba encoder comprises 24 stacked layers, with the output embedding dimension set to 256. The feature space of the extracted high-level forensic features F is $\mathbb{R}^{64\times64\times256}$. All experiments are independently conducted in a consistent environment, with an Intel(R) Xeon(R) Platinum 8358P processor and an NVIDIA Tesla A40-48GB GPU.

Quantitative Comparison

We compared IFDMamba with five baseline models across five datasets: Coverage, NIST, CASIA, MISD, and FF++. The detailed results are presented in Table 2. The results for PSCC-Net, MVSS-Net, and IF-OSN were obtained by retraining these models on the CAT-Net training dataset. The top-performing and second-best results are marked in bold and underlined, respectively. Key observations from the experimental results include:

(1) Deep learning-based classification algorithms provide strong detection performance. Among them, IF-OSN performs well on the MISD dataset, while CAT-Net achieves superior results on the CASIA dataset. The recently proposed TruFor demonstrates robust performance on the other three datasets.

(2) IFDMamba outperforms all baseline methods across all metrics on the five test datasets. Compared to the next best method, IFDMamba demonstrates an average improvement of 5.25% in F1 score and 15.38% in IoU. These substantial gains over baseline models can be attributed to three primary factors. First, IFDMamba leverages a context-aware Mamba architecture to enhance local contextual relationships between image patches while effectively capturing the global contextual relationships across the entire sequence of image patches. This enables the complementary extraction and integration of local fine-grained features and global background features. Second, the use of an improved NT-Xent contrastive loss facilitates pixel-wise contrastive learning, supervising the extraction of high-level forensic features on an image-by-image basis. This explicitly models the relative nature of pristine and forged pixels within a single image—a critical aspect often neglected by existing approaches, which often merge forged or pristine regions from different images into a single category, leading to suboptimal detection performance. Finally, during the testing phase, IFDMamba employs a K-means clustering algorithm to map the extracted high-level forensic features to predicted forgery masks in real-time. This reduces cross-image interference and further enhances the model's generalization capability.

Table 2. Quantitative comparison of different methods on F1 and IoU metrics

Method	**Coverage**		**NIST**		**CASIA**		**MISD**		**FF++**	
	F1	**IoU**	**F1**	**IoU**	**F1**	**IoU**	**F1**	**IoU**	**F1**	**IoU**
PSCC-Net	0.581	0.177	0.629	0.250	0.751	0.472	0.734	0.402	0.513	0.065
MVSS-Net	0.687	0.353	0.634	0.252	0.772	0.513	0.761	0.448	0.632	0.239
IF-OSN	0.653	0.316	0.609	0.226	0.825	0.551	0.767	0.525	0.612	0.223
CAT-Net	0.617	0.233	0.618	0.227	0.842	0.637	0.663	0.313	0.531	0.093
TruFor	0.743	0.452	0.687	0.341	0.831	0.623	0.745	0.421	0.817	0.566
IFDMamba	**0.772**	**0.526**	**0.715**	**0.407**	**0.867**	**0.711**	**0.859**	**0.642**	**0.844**	**0.607**

where TP_x, FP_x, and TN_x denote the true positives, false positives, and true negatives, respectively, for a given class x ("pristine" or "forged").

Qualitative Comparison

Figure 3 illustrates the forgery detection outcomes for selected representative test images. It is evident that PSCC-Net, MVSS-Net, and CAT-Net perform poorly on the test data, failing to detect most of the forged regions while incorrectly identifying some pristine regions as forged. Similarly, IF-OSN misses a significant portion of forged regions. TruFor shows slight improvement in certain cases but still struggles to accurately identify many forged regions and exhibits a noticeable number of false positives.

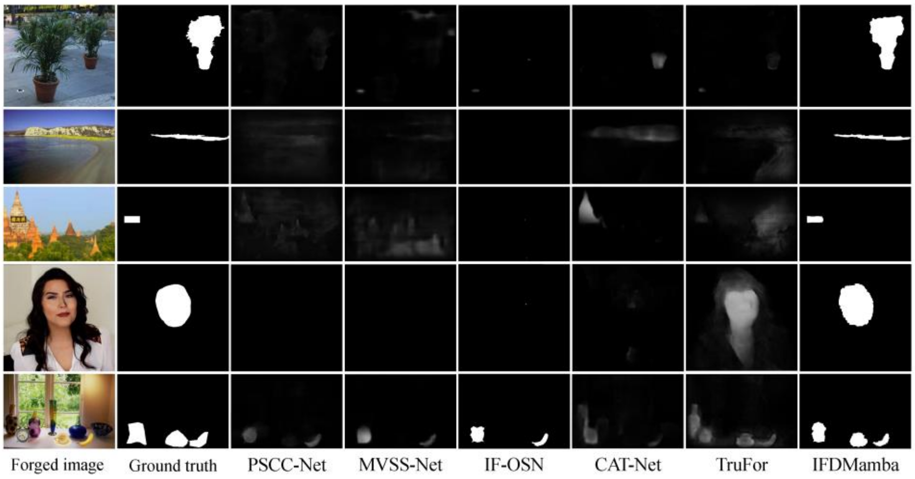


Figure 3. Qualitative comparison of representative image forgery detection

In contrast, IFDMamba effectively identifies forged regions while substantially minimizing the number of false alarms, demonstrating superior performance. The impressive performance of IFDMamba can be attributed to its innovative context-aware Mamba encoder, which enhances local contextual relationships between image patches through the GaSC module and effectively captures global contextual information via the bidirectional modeling mechanism. Additionally, the improved NT-Xent contrastive loss considers all positive keys, explicitly modeling the relative nature of pristine and forged pixels within a single image. This combination of techniques enables IFDMamba to accurately localize forged regions in complex backgrounds, improving both detection accuracy and robustness.

Notably, the last row of Figure 3 presents an example from the MISD dataset, which features multi-source spliced forgeries. Despite the complexity of these forgeries, IFDMamba achieves commendable detection results. This success may stem from IFDMamba's underlying assumption that all forged regions within a single image exhibit similar features, enabling the model to effectively detect multiple types of forgeries simultaneously.

Ablation Studies

Impact of context-aware mamba encoder bidirectional modeling and GaSC module

One distinctive characteristic of the context-aware Mamba encoder is the integration of the GaSC module and the introduction of a bidirectional modeling mechanism. To evaluate the effectiveness of these innovative components, several variants of the context-aware Mamba encoder were designed and tested across five datasets using the F1 score as the evaluation metric. The specific configurations of the variants are as follows:

Variant a (Unidirectional Mamba + Conv1d): This variant employs the Mamba model with a Conv1d convolution layer added before it, processing the image patch sequence in a forward direction only.

Variant b (Bidirectional Mamba + Conv1d): Building upon Variant a, this variant includes an additional pair of Conv1d convolution layers and Mamba models to process the image patch sequence in the backward direction as well.

Variant c (Bidirectional Mamba + GaSC): In this variant, the Conv1d layers in Variant b are replaced with the GaSC module.

The experimental findings in Table 3 reveal several key insights:

Table 3. Impact of bidirectional modeling mechanism and GaSC module

Variants	**Testing Datasets (F1 criterion)**				
	Coverage	**NIST**	**CASIA**	**MISD**	**FF++**
a	0.762	0.705	0.862	0.851	0.837
b	0.769	0.711	0.865	0.856	0.842
c	**0.772**	**0.715**	**0.867**	**0.859**	**0.844**

(1) Variant b outperforms Variant a. This improvement can likely be attributed to the introduction of the bidirectional modeling mechanism, which effectively captures the global contextual relationships across the entire sequence of image patches. In contrast, unidirectional processing in Variant a only captures forward information, failing to comprehensively represent the overall characteristics of forged images.

(2) Variant c achieves the best performance in the experiments. This result may be due to the incorporation of the GaSC module, which enhances the local contextual relationships between image patches. When combined with the bidirectional modeling mechanism, this approach enables effective extraction and complementary integration of local fine-grained features and global background features in forged images, leading to superior detection accuracy.

Impact of improved NT-Xent contrastive loss

Contrastive learning is a critical component of IFDMamba. To investigate the impact of the improved NT-Xent contrastive loss on model performance, comparative experiments were conducted across five datasets using the F1 metric. Detailed outcomes are provided in Table 4.

Table 4. Impact of the improved NT-Xent contrastive loss

Variants	**Testing Datasets (F1 criterion)**				
	Coverage	**NIST**	**CASIA**	**MISD**	**FF++**
NT-Xent	0.768	0.714	0.866	0.852	0.841
Improved NT-Xent	**0.772**	**0.715**	**0.867**	**0.859**	**0.844**

The results indicate that the improved NT-Xent contrastive loss achieves superior detection performance. This improvement is likely because the improved NT-Xent loss considers all positive keys during each loss computation, rather than matching a single positive key. By calculating the expected value of the dot product between the query q and a set of $\{k_i^+\}$, this approach more comprehensively captures the relative nature of pristine and forged pixels within a single image. Consequently, it enhances the accuracy and robustness of detecting forged regions.

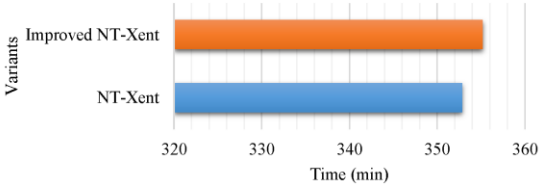


Figure 4. The impact of different loss functions on training time per epoch

Additionally, to assess how various loss functions influence training time, we compared the time required for a single training epoch. As shown in the experimental results in Figure 4, the improved NT-Xent contrastive loss incurs only a marginal increase in computation time, with the additional overhead remaining within an acceptable range. Considering both model performance and training time, the model strikes a balanced compromise between detection efficiency and precision.

CONCLUSION

We propose IFDMamba, a context-aware Mamba-based image forgery detection method. Based on extensive experimental validation, the following conclusions can be drawn:

(1) The proposed context-aware Mamba effectively captures both local contextual relationships between image patches and global contextual information by combining the GaSC module and bidirectional modeling mechanism. This enables efficient and complementary extraction of local fine-grained features and global background features, significantly enhancing the detection capability in complex backgrounds.

(2) The improved NT-Xent contrastive loss explicitly models the relative nature of pristine and forged pixels within a single image through pixel-wise contrastive learning, thereby improving the precision and robustness of forgery region detection.

(3) Experiments on five public datasets—Coverage, NIST, CASIA, MISD, and FF++—demonstrate the superior performance of IFDMamba in image forgery detection tasks, outperforming mainstream methods such as TruFor and CAT-Net.

Future research directions include further optimizing the model to enhance its robustness against noise and artifacts. Additionally, exploring more efficient feature extraction and clustering methods could further improve detection performance and efficiency.

ACKNOWLEDGEMENTS

This research was funded by the National Natural Science Foundation of China, grant number U2268217.

REFERENCES

[1] T. Carvalho, F. A. Faria, H. Pedrini, R. S. Torres, and A. Rocha, “Illuminant-based transformed spaces for image forensics,” IEEE Trans. Inf. Forensics Secur., vol. 11, pp. 720–733, May 2016. doi: 10.1109/TIFS.2015.2506548.

[2] Y. Fan, P. Carré, and C. Fernandez-Maloigne, “Image splicing detection with local illumination estimation,” in Proc. 2015 IEEE Int. Conf. Image Process. (ICIP), pp. 2940–2944, Sep. 2015. doi: 10.1109/ICIP.2015.7351341.

[3] P. Ferrara, T. Bianchi, A. De Rosa, and A. Piva, “Image forgery localization via fine-grained analysis of CFA artifacts,” IEEE Trans. Inf. Forensics Secur., vol. 7, pp. 1566–1577, Nov. 2012. doi: 10.1109/TIFS.2012.2202227.

[4] B. Mahdian and S. Saic, “Using noise inconsistencies for blind image forensics,” Image Vis. Comput., vol. 27, pp. 1497–1503, Nov. 2009. doi: 10.1016/j.imavis.2009.02.001.

[5] C. Dong, X. Chen, R. Hu, J. Cao, and X. Li, “MVSS-Net: Multi-view multi-scale supervised networks for image manipulation detection,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 45, pp. 3539–3553, Apr. 2023.

[6] X. Liu, Y. Liu, J. Chen, and X. Liu, “PSCC-Net: Progressive spatio-channel correlation network for image manipulation detection and localization,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, pp. 7505–7517, Dec. 2022.

[7] H. Wu, J. Zhou, J. Tian, J. Liu, and Y. Qiao, “Robust image forgery detection against transmission over online social networks,” IEEE Trans. Inf. Forensics Secur., vol. 17, pp. 443–456, Jan. 2022. doi: 10.1109/TIFS.2022.3144878.

[8] F. Guillaro, D. Cozzolino, A. Sud, N. Dufour, and L. Verdoliva, “TruFor: Leveraging all-round clues for trustworthy image forgery detection and localization,” in Proc. 2023 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 20606–20615, Jun. 2023.

[9] M. Huh, A. Liu, A. Owens, and A. A. Efros, “Fighting fake news: Image splice detection via learned self-consistency,” in Computer Vision – ECCV 2018: 15th European Conf., Munich, Germany, Sept. 8–14, 2018, pp. 106–124.

[10] Y. Li and J. Zhou, “Fast and effective image copy-move forgery detection via hierarchical feature point matching,” IEEE Trans. Inf. Forensics Secur., vol. 14, pp. 1307–1322, May 2019. doi: 10.1109/TIFS.2018.2876837.

[11] H. Wu and J. Zhou, “IID-Net: Image inpainting detection network via neural architecture search and attention,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, pp. 1172–1185, Mar. 2022. doi: 10.1109/TCSVT.2021.3075039.

[12] M.-J. Kwon, S.-H. Nam, I.-J. Yu, H.-K. Lee, and C. Kim, “Learning JPEG compression artifacts for image manipulation detection and localization,” Int. J. Comput. Vis., vol. 130, pp. 1875–1895, Dec. 2022.

[13] A. Gu, K. Goel, and C. Ré, “Efficiently modeling long sequences with structured state spaces,” arXiv e-prints, 2021, arXiv:2111.00396.

[14] A. Gu, I. Johnson, K. Goel, K. Saab, T. Dao, A. Rudra, et al., “Combining recurrent, convolutional, and continuous-time models with linear state space layers,” in Advances in Neural Information Processing Systems, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, and J. W. Vaughan, Eds., Curran Associates, Inc., 2021, pp. 572–585.

[15] J. T. H. Smith, A. Warrington, and S. W. Linderman, “Simplified state space layers for sequence modeling,” arXiv e-prints, 2022, arXiv:2208.04933.

[16] A. Gu, T. Dao, S. Ermon, A. Rudra, and C. Ré, “HiPPO: Recurrent memory with optimal polynomial projections,” in Advances in Neural Information Processing Systems, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., Curran Associates, Inc., 2020, pp. 1474–1487.

[17] A. Gu and T. Dao, “Mamba: Linear-time sequence modeling with selective state spaces,” arXiv e-prints, 2023, arXiv:2312.00752.

[18] E. Nguyen, K. Goel, A. Gu, G. Downs, P. Shah, T. Dao, et al., “S4ND: Modeling images and videos as multidimensional signals with state spaces,” in Advances in Neural Information Processing Systems, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., Curran Associates, Inc., 2022, pp. 2846–2861.

[19] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, et al., "An image is worth 16x16 words: Transformers for image recognition at scale," 2021.

[20] J. N. Yan, J. Gu, and A. M. Rush, "Diffusion models without attention," arXiv e-prints, 2023, arXiv:2311.18257.

[21] E. Martin and C. Cundy, "Parallelizing linear recurrent neural nets over sequence length," arXiv e-prints, 2017, arXiv:1709.04057. doi: 10.48550/arXiv.1709.04057.

[22] Q. Yin, J. Wang, W. Lu, and X. Luo, "Contrastive learning based multi-task network for image manipulation detection," Signal Process., vol. 201, p. 108709, Nov. 2022.

[23] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in Proc. 2018 IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), Los Alamitos, CA, USA, Jun. 2018, pp. 3733–3742. doi: 10.1109/CVPR.2018.00393.

[24] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in Proc. 37th Int. Conf. Machine Learning, H. D. III and A. Singh, Eds., PMLR, 2020, pp. 1597–1607.

[25] N. F. Niloy, K. Kumar Bhaumik, and S. S. Woo, "CFL-Net: Image forgery localization using contrastive learning," in Proc. 2023 IEEE/CVF Winter Conf. Applications of Computer Vision (WACV), Jan. 2023, pp. 4631–4640. doi: 10.1109/WACV56688.2023.00462.

[26] B. Wen, Y. Zhu, R. Subramanian, T.-T. Ng, X. Shen, and S. Winkler, "COVERAGE — A novel database for copy-move forgery detection," in Proc. 2016 IEEE Int. Conf. Image Processing (ICIP), 2016, pp. 161–165. doi: 10.1109/ICIP.2016.7532339.

[27] H. Guan, M. Kozak, E. Robertson, Y. Lee, A. N. Yates, A. Delgado, et al., "MFC datasets: Large-scale benchmark datasets for media forensic challenge evaluation," in Proc. 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW), 2019, pp. 63–72.

[28] J. Dong, W. Wang, and T. Tan, "CASIA image tampering detection evaluation database," in Proc. 2013 IEEE China Summit and Int. Conf. Signal and Information Processing, 2013, pp. 422–426. doi: 10.1109/ChinaSIP.2013.6625374.

[29] K. D. Kadam, S. Ahirrao, and K. Kotecha, "Multiple image splicing dataset (MISD): A dataset for multiple splicing," Data, vol. 6, p. 102, Oct. 2021. doi: 10.3390/data6100102.

[30] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Niessner, "FaceForensics++: Learning to detect manipulated facial images," in Proc. 2019 IEEE/CVF Int. Conf. Computer Vision (ICCV), Los Alamitos, CA, USA, Oct. 2019, pp. 1–11. doi: 10.1109/ICCV.2019.00009.

[31] N. T. Pham, J. W. Lee, G. R. Kwon, and C. S. Park, "Hybrid image-retrieval method for image-splicing validation," Symmetry, vol. 11, p. 83, Jan. 2019. doi: 10.3390/sym11010083.