symmetry

MDPI

Article

Enhancing Document Forgery Detection with Edge-Focused Deep Learning

Yong-Yeol Bae, Dae-Jea Cho * and Ki-Hyun Jung *

Department of Software Convergence, Gyeongkuk National University (Andong National University), Andong 36729, Republic of Korea; baeyongyeol@gmail.com
* Correspondence: djcho@anu.ac.kr (D.-J.C.); kingjung@gknu.ac.kr (K.-H.J.)

Abstract

Detecting manipulated document images is essential for verifying the authenticity of official records and preventing document forgery. However, forgery artifacts are often subtle and localized in fine-grained regions, such as text boundaries or character outlines, where visual symmetry and structural regularity are typically expected. These manipulations can disrupt the inherent symmetry of document layouts, making the detection of such inconsistencies crucial for forgery identification. Conventional CNN-based models face limitations in capturing such edge-level asymmetric features, as edge-related information tends to weaken through repeated convolution and pooling operations. To address this issue, this study proposes an edge-focused method composed of two components: the Edge Attention (EA) layer and the Edge Concatenation (EC) layer. The EA layer dynamically identifies channels that are highly responsive to edge features in the input feature map and applies learnable weights to emphasize them, enhancing the representation of boundary-related information, thereby emphasizing structurally significant boundaries. Subsequently, the EC layer extracts edge maps from the input image using the Sobel filter and concatenates them with the original feature maps along the channel dimension, allowing the model to explicitly incorporate edge information. To evaluate the effectiveness and compatibility of the proposed method, it was initially applied to a simple CNN architecture to isolate its impact. Subsequently, it was integrated into various widely used models, including DenseNet121, ResNet50, Vision Transformer (ViT), and a CAE-SVM-based document forgery detection model. Experiments were conducted on the DocTamper, Receipt, and MIDV-2020 datasets to assess classification accuracy and F1-score using both original and forged text images. Across all model architectures and datasets, the proposed EA–EC method consistently improved model performance, particularly by increasing sensitivity to asymmetric manipulations around text boundaries. These results demonstrate that the proposed edge-focused approach is not only effective but also highly adaptable, serving as a lightweight and modular extension that can be easily incorporated into existing deep learning-based document forgery detection frameworks. By reinforcing attention to structural inconsistencies often missed by standard convolutional networks, the proposed method provides a practical solution for enhancing the robustness and generalizability of forgery detection systems.

Keywords: digital document; forgery detection; edge; deep learning; CNN

Academic Editor: Zhixun Su

Received: 30 May 2025
Revised: 3 July 2025
Accepted: 17 July 2025
Published: 30 July 2025

Citation: Bae, Y.-Y.; Cho, D.-J.; Jung, K.-H. Enhancing Document Forgery Detection with Edge-Focused Deep Learning. *Symmetry* **2025**, *17*, 1208. https://doi.org/10.3390/sym17081208

1. Introduction

With the rapid advancement of multimedia technology, the practice of digitizing documents across various fields has greatly increased. Managing critical information through document images offers several advantages in terms of information dissemination and storage efficiency. This approach not only overcomes temporal and spatial constraints but also helps save resources required for printing. Consequently, many companies and government agencies now archive documents as image files and distribute them as needed, enabling more efficient information management. However, the ease of advanced image editing has led to a rise in sophisticated document image forgeries, which can cause significant societal harm [1].

To detect such forgeries, extensive research has been conducted, broadly categorized into traditional methods and deep learning-based methods. Traditional forgery detection techniques typically rely on handcrafted features (e.g., Local Binary Patterns (LBP), Scale-Invariant Feature Transform (SIFT), and Discrete Cosine Transform (DCT)). These methods analyze clues such as pixel-level anomalies, format inconsistencies, or geometric misalignments to determine whether an image has been tampered with. Traditional approaches are generally computationally efficient, but they may be limited in dealing with very sophisticated manipulations or new types of attacks. In contrast, deep learning-based approaches—including Convolutional Neural Networks (CNNs), Fully Convolutional Networks (FCNs), and Generative Adversarial Network (GAN)-based detectors—have demonstrated superior generalization performance, enabling more effective detection of complex forgeries (such as Deepfakes).

However, these approaches often require large training datasets, long training times, and substantial computational resources [2]. To alleviate these drawbacks, recent studies have introduced larger datasets and more efficient models. For example, Mahfoudi et al. introduced the Copy-Move ID (CMID) dataset, which consists of 893 copy-move forged ID document images and 304 authentic document images, along with corresponding ground-truth masks [3]. Notably, this dataset contains numerous repeating characters or Similar but Genuine Objects (SGOs) and extremely small tampered-with regions, posing a significant challenge for existing detection algorithms. In another recent study, Li et al. proposed a document image forgery detection model that integrates spatial-frequency feature fusion with a multi-scale network [4]. Their method leverages frequency–domain representations (DCT) together with an HRNet-based multi-scale feature extractor to enhance the accurate localization of forged regions. The model further employs an attention mechanism and a multi-category forgery-classification head, achieving high detection performance across a variety of forgery types.

Despite these advancements, existing deep learning-based methods, including those by Mahfoudi et al. [3] and Li et al. [4], share common structural limitations. These methods often implicitly handle edge-related structural features and are typically bound to specific architectures, limiting their sensitivity to subtle structural inconsistencies around character boundaries and edges. This limitation is critical, as changes in edge information are primary indicators of forgery in document images [5].

The importance of edge cues in document image forensics has been emphasized in prior studies. For instance, Qu et al. [6] proposed the Global view and Edge Attention Module (GEAM), which integrates edge features into an attention mechanism to improve the detection of tampered-with regions obscured by blending artifacts. Their work highlights how boundary-level distortions, often invisible to texture-based analysis, can be effectively revealed through edge-enhanced representations. Building on this insight, our study further explores the role of edge information by introducing two explicitly designed and lightweight modules: the Edge Attention (EA) layer and the Edge Concatenation (EC)

layer. These modules leverage Sobel-based edge extraction to dynamically emphasize edge-responsive channels (EA) and directly inject edge maps into the network's representation space (EC), thereby enhancing the model's sensitivity to subtle structural inconsistencies. Unlike methods like GEAM, which operate within a fixed architecture, the EA and EC layers are designed to be easily integrated into a wide variety of deep learning models—ranging from general-purpose CNNs like ResNet50 and DenseNet121 to task-specific detectors—without structural modifications. This flexibility allows broader applicability and modular evaluation.

This study directly addresses the structural limitations of previous approaches by introducing two novel components—the Edge Attention (EA) and Edge Concatenation (EC) layers—specifically designed to capture and enhance subtle edge-related structural features. The key contributions of this research are as follows:

- The design and implementation of edge-focused layers (EA and EC) with high modularity, enabling explicit extraction and emphasis of fine-grained edge information.
- Comprehensive experimental validation demonstrating consistent improvements in detection accuracy and robustness across multiple benchmark datasets.

The remainder of this paper is organized as follows: Section 2 reviews related work; Section 3 presents our proposed methodology; Section 4 details experimental results; and Section 5 concludes the paper.

2. Related Work

2.1. Document Image Forgery Techniques

Forgery techniques in document images generally fall into three main categories: copy-move, splicing, and insertion. These techniques manipulate document content by either duplicating parts within the same file, incorporating elements from other sources, or artificially introducing new content. The main forgery techniques and their details are summarized in Table 1.

Table 1. Techniques of document forgery.

Forgery Type	Details
Copy-Move	This technique replicates a portion of content—either text or image—from one region of a document and relocates it to another area within the same document. It is frequently employed to obscure or duplicate specific information, such as repeating a signature or value to mislead the reader. By maintaining consistent visual patterns, such manipulations can be difficult to detect through simple visual inspection [7].
Splicing	Splicing refers to the integration of external elements—sourced from different documents—into the target document. Unlike copy-move, this approach introduces foreign content, which may serve to create false scenarios or replace original information. For example, transplanting a signature from another source can create a falsified but seemingly authentic document [7].
Insertion	Insertion involves the deliberate addition of new characters or graphical elements using digital editing tools. This method is typically used to subtly alter critical details such as dates, figures, or names. As editing software evolves, insertion-based forgery has become more refined and challenging to detect, prompting the need for specialized detection strategies [8].

2.2. The Traditional Document Forgery Detection Methods

Traditional forgery detection techniques can be broadly categorized into block-based methods, key-point-based methods, and noise analysis-based methods. Block-based methods divide an image into small overlapping blocks and compare the features of each block to identify duplicated or altered regions. For example, Discrete Cosine Transform (DCT) was applied to each block to extract frequency–domain features, which were then sorted and blocks with identical offsets were identified in [9]. Similarly, Popescu et al. [10] proposed a method that extracts block-level features, reduces dimensionality using Principal Component Analysis (PCA), and matches similar blocks efficiently. Key-point-based methods detect forgery by extracting distinctive key-points that are invariant to rotation or scaling, and then matching these points to identify tampered-with regions. In [11], Scale-Invariant Feature Transform (SIFT) features were used to detect copy-move forgeries under geometric transformations. Additionally, Christlein et al. [12] evaluated various copy-move forgery detection methods based on Speeded-Up Robust Features (SURF), comparing their performance under different post-processing operations. Noise-based approaches analyze statistical noise characteristics across different regions of an image. A representative method in [13] utilized inconsistencies in local noise variances to detect spliced regions, leveraging the subtle noise irregularities resulting from merging fragments from different sources.

2.3. Deep Learning-Based Forgery Detection Methods

Recently, with the advancement of deep learning technologies, data-driven approaches have been introduced in the field of document forgery detection. Unlike traditional methods that rely on hand-crafted features, deep learning-based techniques automatically learn subtle traces of forgery from large-scale training data, often achieving higher detection performance. For example, Gornale et al. [8] proposed a deep learning-based model that detects document image forgery using RGB color channels. Their method extracts features separately from each color channel (R, G, B) to highlight the differences between genuine and forged documents, and feeds them into a convolutional neural network (CNN) for classification. The experimental results showed that this color channel-based approach is particularly effective in detecting forgeries in color documents compared to grayscale ones. Similarly, Nandanwar et al. [14] introduced a novel deep learning framework to detect altered text in document images. Their model first accurately localizes text regions, then employs a CNN architecture to analyze visual inconsistencies between tampered-with and untampered-with areas. This method demonstrated high detection accuracy even in challenging conditions such as low-resolution or complex backgrounds, making it well-suited for text-centric forgery detection. Additionally, Tyagi et al. [15] proposed an unsupervised deep learning approach that simultaneously performs forgery detection and writer identification. Using an autoencoder without any prior labeling, the model learns latent features of document images, enabling it to detect forgeries as well as identify the unique writing style of the author. The results confirmed its robust performance under various forgery scenarios and its potential applicability in real-world document forensics.

2.4. The Sobel Operator

The Sobel operator is one of the most widely used classical edge detection techniques for extracting edge information in images. It estimates the direction and magnitude of edges at each pixel by computing discrete derivatives of the image intensity function. According to Zhang et al. [16], the Sobel operator approximates the gradient in the horizontal and vertical directions using a 3×3 neighborhood centered around each pixel, assigning higher

weights to the center to enhance robustness against noise. The gradients in the x and y directions are computed using the convolution operations defined in Equation (1).

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \; S_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \tag{1}$$

Applying these two kernels to the input image yields the horizontal gradient S_x and the vertical gradient S_y, from which the edge magnitude G at each pixel is calculated as shown in Equation (2).

$$S = \sqrt{S_x^2 + S_y^2} \tag{2}$$

Alternatively, a more computationally efficient approximation, as shown in Equation (3), may also be employed.

$$S \approx |S_x| + |S_y| \tag{3}$$

Zhang et al. [16] evaluated the Sobel operator as an effective edge detection method due to its relatively low computational cost, ease of implementation, and noise suppression characteristics through central weighting. Moreover, because it responds sensitively to edges in both the horizontal and vertical directions, it is considered useful in the initial stages of edge extraction. Its ability to emphasize directional changes makes it particularly advantageous for preprocessing tasks in computer vision pipelines, where reliable edge detection is crucial for downstream applications such as segmentation, recognition, and feature extraction.

3. The Proposed Method

In this section, we propose an edge-based feature extraction method for document forgery detection. The core idea is to explicitly integrate edge information into a deep neural network, enabling the model to focus on sharp discontinuities and boundary artifacts that may occur in forged text. To achieve this, we introduce two novel modular layers: the Edge Attention Layer (EA) and the Edge Concatenation Layer (EC). These modules can be inserted into standard convolutional architectures such as DenseNet121 [17], ResNet50 [18], ViT [19], or custom CNNs to enhance the representation of edge-related regions. The Edge Attention Layer (EA) enhances channels that are highly responsive to edge information through an edge-based channel attention mechanism. In contrast, the Edge Concatenation Layer (EC) directly concatenates edge feature maps—extracted using the Sobel filter—along the channel dimension, allowing the CNN to explicitly utilize edge information.

By leveraging such edge information, the network can more effectively focus on unnatural boundaries that are commonly observed in forged documents, such as irregular character outlines or pasted region edges.

Edges reflect sudden changes in intensity and serve as key indicators for separating different regions in an image. In authentic document images, edge patterns usually follow expected structures, such as printed text boundaries or geometric lines. However, forgery operations often disrupt these regularities and introduce unnatural gradients and boundary discontinuities, which can be effectively captured through edge analysis.

As noted by Macit and Koyun [20], edge data is one of the most crucial sources of structural detail in an image and can be effectively utilized to distinguish between original and tampered-with regions, especially when properly integrated into feature extraction or model design. Nevertheless, in conventional CNN architectures, important feature information is often lost through repeated pooling and convolution operations [21], which reduce spatial resolution and may suppress subtle visual cues. Existing approaches such as Squeeze-and-Excitation (SE) blocks [22] and HRNet [23] attempt to address the limitations

of feature degradation in CNNs through different strategies. SE blocks enhance feature representations by learning channel-wise importance weights from global average pooled features. However, this mechanism operates on holistic feature statistics and does not explicitly consider localized edge structures that are crucial for detecting boundary-level forgeries. HRNet, on the other hand, maintains high-resolution representations via multi-branch processing and repeated fusion of different scales. Although this design preserves spatial detail, it does not specifically isolate or emphasize edge-related artifacts introduced by manipulation.

In contrast, our proposed Edge Attention (EA) layer directly measures channel sensitivity to edge responses—computed from Sobel filters—and selectively amplifies channels that strongly react to edge content. The Edge Concatenation (EC) layer explicitly injects edge maps into the feature space by concatenating them with the original input, enabling the network to learn directly from edge cues. Together, these modules provide a targeted and modular solution that explicitly incorporates edge information, making them fundamentally different from existing refinement or fusion strategies.

3.1. The Architecture of the Edge Attention Layer

3.1.1. Overview of the Edge Attention Layer

The Edge Attention Layer (EA) is designed to dynamically assign weights to each channel based on the edge information extracted from the input convolutional feature map.

In a typical CNN, multiple feature maps are generated, but only a subset of them may strongly respond to edge patterns that are useful for forgery detection, such as those highlighting character boundaries or object contours.

The Edge Attention Layer (EA) identifies and emphasizes the channels with prominent edge responses, while leaving the others unchanged or less emphasized.

This mechanism is similar to channel attention modules; however, unlike the conventional Squeeze-and-Excitation (SE) approach, which relies on learned global statistics, the EA layer uses edge magnitudes calculated via Sobel operators as its attention signal.

By focusing on the top K channels with the strongest edge activations, the network is guided to better capture subtle signs of manipulation, such as distorted text outlines or splice boundaries. Here, K controls the number of edge-responsive channels selected for emphasis. In our experiments, K was set to 3.

3.1.2. Structure and Operation

In this section, we describe the operation of the EA layer step by step.

Step 1. Per-Channel Edge Extraction

For each channel F_i of the input feature map $F \in R^{C\times H\times W}$, horizontal and vertical gradients are computed using the Sobel filters S_x, S_y as defined in Equation (1).

The gradient computation for each channel is given by Equation (4):

$$G_{x,i} = F_i * S_x, \quad G_{y,i} = F_i * S_y \tag{4}$$

Then, the edge magnitude at each spatial location is calculated as in Equation (5):

$$M_i(x,y) = \sqrt{G_{x,i}(x,y)^2 + G_{y,i}(x,y)^2} \tag{5}$$

As a result, an edge map $M_i \in R^{H\times W}$ is obtained for each channel.

Step 2. Channel-Wise Edge Strength Calculation

The average value of each edge map M_i, generated in Step 1, is computed to obtain a scalar value E_i representing the edge strength of the corresponding channel.

This calculation is defined in Equation (6):

$$E_i = \frac{1}{H \times W} \sum_{x=1}^{H} \sum_{y=1}^{W} M_i(x, y) \tag{6}$$

The resulting vector $E = [E_1, E_2, E_3, \ldots, E_c]$ represents the edge strength of all channels.

Step 3. Top-K Channel Selection

The indices of the top K channels with the highest edge strength values are selected, as defined in Equation (7):

$$J = argsort_K(E) \tag{7}$$

Here, J denotes the set of indices corresponding to the K channels with the largest values in E.

Step 4. Learned Weighting Mechanism

A learnable weight α_{edge} is applied to the selected top K channels to emphasize their importance. The parameter α_{edge} was initialized to 1.5 and is updated through backpropagation along with the rest of the model, and optimized using the Adam optimizer based on gradients computed from the binary cross-entropy loss. Through this process, the network learns—based on the training data—how much emphasis should be placed on edge-dominant channels to improve performance. The weight factors w_i for each channel i are defined as in Equation (8):

$$w_i = \begin{cases} 1 + \alpha_{edge}, & if\ i \in J\ 1, \quad otherwise \end{cases} \tag{8}$$

Step 5. Channel Reweighting

Each channel F_i is multiplied by its corresponding weight w_i to obtain the final output F_i', as defined in Equation (9):

$$F_i'(x, y) = w_i F_i(x, y) \quad for\ i = 1, \ldots, C \tag{9}$$

The final output tensor $F' \in R^{C \times H \times W}$ represents the feature map after applying edge-based channel attention, and is passed to the subsequent layers of the network.

Through this process, the EA guides the network's focus toward edge-responsive features that are more likely to be associated with forgery, thereby contributing to improved performance in document image forgery detection.

3.2. The Architecture of the Edge Concatenation Layer

3.2.1. Overview of the Edge Concatenation Layer

The Edge Concatenation Layer (EC) is a module designed to enable the neural network to explicitly utilize edge information. This layer applies the Sobel operator to the input feature map to generate edge maps, which are then concatenated along the channel dimension with the original features, allowing the model to directly receive edge information as part of its input.

While conventional CNNs are expected to learn edge patterns implicitly through internal filters, the Edge Concatenation Layer (EC) explicitly provides handcrafted edge features—calculated using domain knowledge via the Sobel filter—thus clearly delivering edge-specific information to the network.

In the context of document forgery detection, the proposed method module introduces explicit edge cues at the input stage, supporting the model in more reliably learning high-frequency characteristics, such as boundaries of tampered-with regions. Furthermore, since the edge maps are perfectly aligned with the original features at the pixel level, subse-

quent convolutional layers can jointly analyze texture and edge information at the same spatial locations.

3.2.2. Structure and Operation

In this section, we describe the operation of the EC layer step by step.

Step 1. Edge Extraction per Channel

In the operation of the EA layer, Step 1 and Step 2—corresponding to Equations (4) through (6)—are performed to compute the edge map set $E = [E_1, E_2, E_3, \ldots, E_c]$, where each E_i represents the edge map of the i-th channel.

Step 2. Concatenation (Channel-wise Merging)

The core operation of the EC is to concatenate the computed edge tensor E with the original input tensor X along the channel dimension (i.e., axis = -1). This operation produces an expanded feature tensor $X' \in R^{(C+C_e)\times H\times W}$, where C denotes the number of channels in the original input features, and C_e represents the number of edge channels computed using the Sobel filter.

This concatenation appends the edge information to the original features, enabling the subsequent layers to explicitly utilize edge cues at each spatial location.

3.3. Effect of Sequential Connection of Edge Attention and Edge Concatenation Layers

In this section, we propose an edge-focused attention module that sequentially integrates the EA layer and the EC layer to jointly perform edge enhancement and feature representation refinement. First, the EA layer applies the Sobel operator to the input feature map to compute the edge strength of each channel, and assigns a learnable weight to the top K edge-responsive channels. This enable the network to better preserve features that are sensitive to edge information. Next, the enhanced feature map is passed to the EC layer, which again applies the Sobel operator to the emphasized representation and concatenates the resulting edge maps with the original features along the channel dimension. This process goes beyond simply adding edge maps; it produces an edge-enhanced representation that is guided by semantically relevant channel selection.

As a result, the combination of these two layers minimizes the loss of high-frequency information and enables the network to more precisely learn fine visual cues, such as tampered-with boundaries or structural discontinuities in forged documents. This two-layer structure is proposed as a single edge-focused module, as shown in Figure 1.

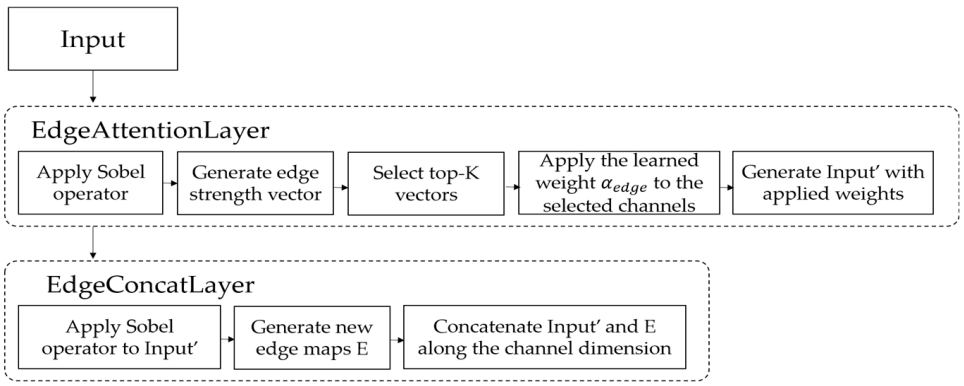


Figure 1. Overall Structure of the EA and EC Layers.

4. Experimental Results

The experimental results are presented in three sections. The first section analyzes how effectively the EA and EC layers emphasize edge regions in a basic CNN model, based on both visual and numerical differences. The second section examines performance

variations when the EA and EC layers are integrated into the same baseline CNN model used in the first section, comparing results across different kernel sizes and numbers of channels. Finally, the third section compares the performance impact of incorporating the EA and EC layers into widely used CNN architectures (such as DenseNet121 and ResNet50) as well as recently proposed deep learning-based document forgery detection models. The basic CNN model architecture used in the first and second sections is illustrated in Figure 2.

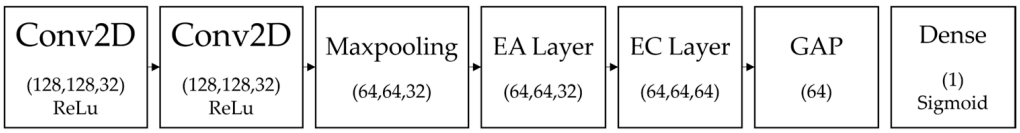


Figure 2. Basic CNN architecture used for evaluating the effectiveness of the EA and EC layers.

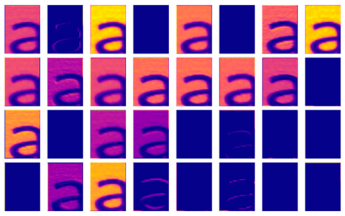
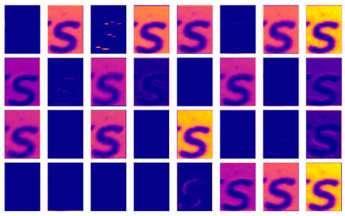
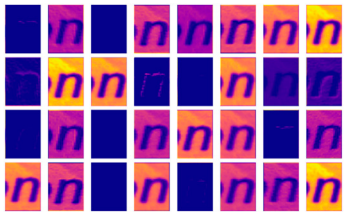
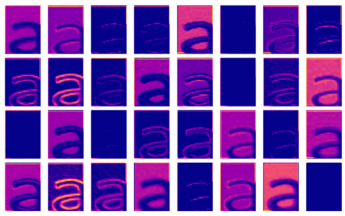
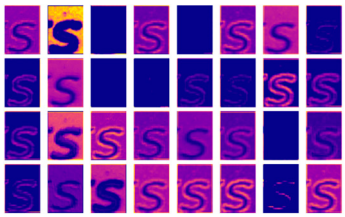
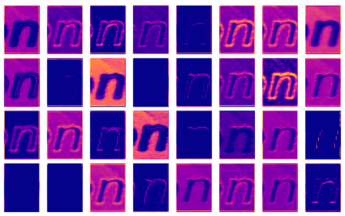
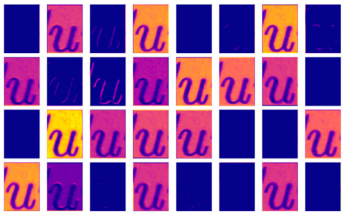
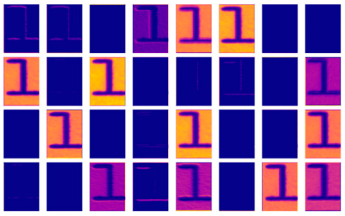
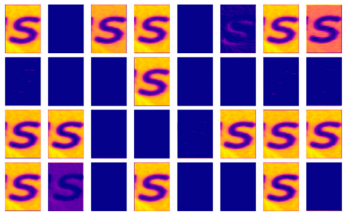
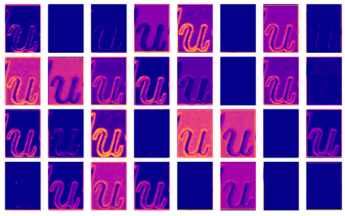
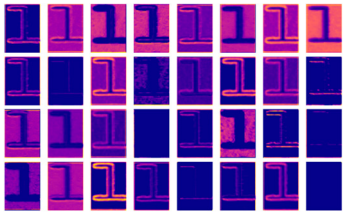
4.1. Overview of Experimental Datasets

To evaluate the performance of the proposed EA and EC layers, three different datasets were used for experimentation. The DocTamper dataset, introduced in [24], consists of 170,000 document images in English and Chinese; in this study, only the English-language images were used. The Receipt dataset [25] was originally used for a document forgery detection competition and contains English receipt images. The MIDV-2020 dataset [26] is a publicly available dataset comprising scanned and photographed identity documents, such as passports, driver's licenses, and ID cards, designed primarily for document analysis and recognition tasks. Since MIDV-2020 is not a forgery dataset, forged images were artificially created using the image editing tool GIMP. The Receipt and MIDV-2020 datasets were manipulated using the copy-move forgery method, while the DocTamper dataset includes both copy-move and splicing forgeries. All experiments in this study were conducted using these three datasets.

4.2. Feature Map and Edge Activation Comparison Through Ablation Study

For this comparison, we prepared two models: a baseline CNN, as shown in Figure 2, consisting of two simple convolutional layers, and a modified version in which the EA and EC layers were inserted between the convolutional layers. From a randomly selected image in the dataset, feature maps were extracted at the output of the final convolutional layer in each model. A total of 32 feature channels were visualized using the plasma colormap, allowing us to qualitatively compare how the proposed modules affect the activation patterns related to edge information. The activation values in the feature maps represent the intensity of response of each convolutional filter to local patterns in the input. After applying ReLU, only positive values are retained, highlighting the regions where the model detects meaningful features. For this experiment, we used the DocTamper dataset introduced in [24], using randomly selected alphabetic characters a, s, n, u, and l from the dataset for analysis. To examine the degree of edge enhancement around text regions, the document images were segmented at the word level using Tesseract OCR, and each segmented word image was used as input to the model. The feature map results of the basic CNN model and the model with the EA and EC layers applied are shown in Table 2. As shown in Table 2, there is a clear visual difference between the feature maps of the basic CNN model and those of the model with the EA and EC layers. In the basic CNN model, edge regions corresponding to the boundaries between text and background are not particularly prominent. In contrast, the model with the EA and EC layers exhibits stronger activation along the text boundaries, making the edges more visually pronounced.

Table 2. Feature map comparison between the baseline CNN and the model with EA and EC layers using word-level inputs from the DocTamper dataset.

		Example A	Example B	Example C
Original text image	CNN			
	CNN + EA, EC layers			
Forged text image	CNN			
	CNN + EA, EC layers			

To examine not only the visual differences but also the numerical differences in edge-specific activation, we used 519 original and 519 forged text images from the DocTamper dataset. Specifically, we calculated and compared the mean activation, maximum activation, and standard deviation within the edge regions. The edge regions were defined using the Sobel operator, where the edge strength E was computed as the sum of the absolute gradients in the x and y directions:

$$E = |G_x| + |G_y| \tag{10}$$

A binary edge mask $M(i,j)$ was then generated by applying a threshold to the edge map:

$$M(i,j) = \{1 \ \ if \ E(i,j) > threshold \, 0 \ \ otherwise \tag{11}$$

where the threshold was set to 10% of the maximum edge strength in each image, i.e., $threshold = 0.1 \times max(E)$. This mask was used to extract edge-focused activations from the CNN feature map A, resulting in a masked activation map:

$$A_{edge} = A \cdot M \tag{12}$$

The mean activation within the edge regions was then calculated as follows:

$$\mu_{edge} = \frac{\sum A_{edge}}{\sum M} \tag{13}$$

Equations (10)–(13) enabled a quantitative evaluation of each model's sensitivity to edge regions, which are critical in distinguishing forged text. In addition, a five-fold cross-validation was conducted to ensure the robustness and consistency of the measurements. The results, presented in Table 3, show that the model incorporating the EA and EC layers consistently produced higher edge-focused activation values, indicating improved sensitivity to boundary-level manipulations characteristic of forged text. This suggests that the edge-enhancement mechanism effectively amplifies subtle structural cues along text boundaries, which are often overlooked in conventional CNNs. As a result, the model becomes better equipped to discriminate between authentic and manipulated content, especially in regions where forgery artifacts are visually subtle.

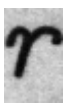
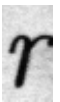
Table 3. Comparison of edge-region activation statistics between the baseline CNN and the CNN with EA and EC layers.

Model	Mean	Max	Std Dev
CNN	2.2226	3.1151	0.2049
CNN + EA, EC layers	3.3556	4.2695	0.3488

To more precisely analyze the impact of the EA and EC layers on the edge regions of original and forged text, we input original and forged text images into a CNN model incorporating the EA and EC layers and calculated the average activation values of the output channels using Equations (10)–(13), with a focus on those corresponding to edge regions. Since activation values can vary depending on the shape of the characters, we ensured fair comparisons by selecting pairs of forged and original text images containing the same alphabetic characters. This experiment used a subset of the DocTamper dataset. The matched alphabet data is presented in Table 4, and the average activation values across channels for each character, both original and forged, are compared in Figure 3.

Beyond the experiment comparing edge-specific mean activation values using the same alphabetic characters, we also conducted a broader analysis across the entire dataset. This experiment was conducted using the DocTamper, Receipt, and MIDV-2020 datasets. To ensure reliable evaluation, we applied five-fold cross-validation when measuring the edge mean activation values of both original and forged text images. The forged samples were manipulated using copy-move and insertion methods, and all measurements followed the same procedure. The final results, averaged over five repetitions, are reported in Table 5.

Table 4. Pairs of original and forged text images containing the same alphabetic characters used for activation comparison.

Alphabet	w	i	t	a	r	o	u	e	n	s
Original Images										
Forged Images										

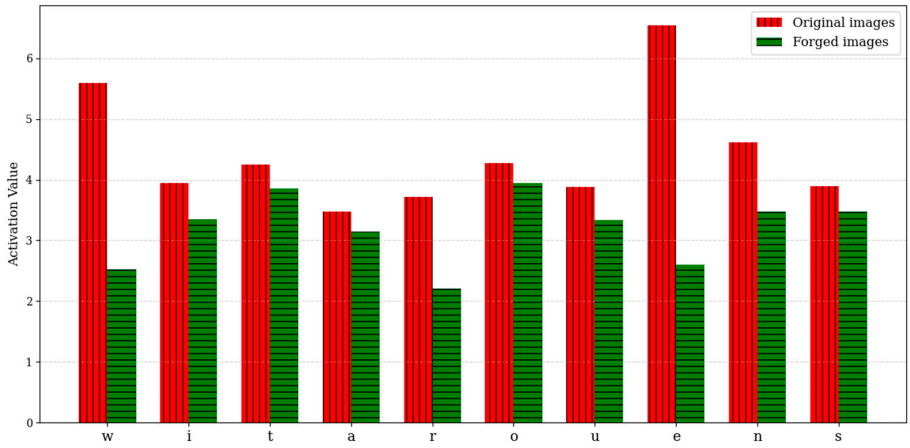


Figure 3. Comparison of mean edge-focused activation values between forged and original text images by character.

Table 5. Comparison of activation statistics between original and forged text images.

Dataset		Mean Activation	Max Activation
DocTamper	Original	3.6723	4.9416
	Forged	3.1567	4.0629
Receipt	Original	3.6097	4.8683
	Forged	3.2155	3.9911
MIDV-2020	Original	3.4899	3.5464
	Forged	1.4950	1.5164

As shown in Figure 3 and Table 5, the original text images generally exhibit higher mean activation values in edge regions than the forged text images. This is likely because the edges in original text images tend to be cleaner and more naturally aligned with the background, resulting in stronger activation responses. In contrast, forged text often contains subtle artifacts or inconsistencies introduced during manipulation, which may weaken the edge response. Based on these results, it can be concluded that the EA and EC layers effectively enhance the edge regions of text images and amplify the differences between original and forged text, which can lead to improved detection performance.

4.3. Ablation Study on EA and EC Layers Using a Basic CNN Architecture

To examine the impact of the EA and EC layers on classification performance in document forgery detection, we compared a basic CNN model composed of two convolutional layers with a modified version that integrates the EA and EC layers into the same base architecture. The reason for using a simple CNN model with only two convolutional layers in this experiment was to isolate the effect of the EA and EC layers as much as possible, minimizing interference from other architectural factors. In Section 4.5, we extend the analysis by incorporating the EA and EC layers into more practical image classification or document forgery detection models to further evaluate their effectiveness. When using the basic CNN model, we assumed that the kernel size and the number of channels in the convolutional layers could affect the model's classification performance. Therefore, experiments were conducted with three different channel sizes (32, 64, and 128) and three kernel sizes (3, 5, and 7) to compare classification accuracy under various configurations. In the experiment, input images with a size of 128 $\times$ 128 and three RGB channels were used. All convolutional layers employed the ReLU activation function, while the output layer used a Sigmoid function for binary classification. The loss function was set to Binary Cross-Entropy, and the model was trained using the Adam optimizer for a total of 10 epochs. Model performance was evaluated using five-fold cross-validation, with Accuracy and

F1-score as the evaluation metrics. The datasets used in the experiments are the DocTamper, Receipt, and MIDV-2020 datasets. The experimental results are shown in Figures 4–6.

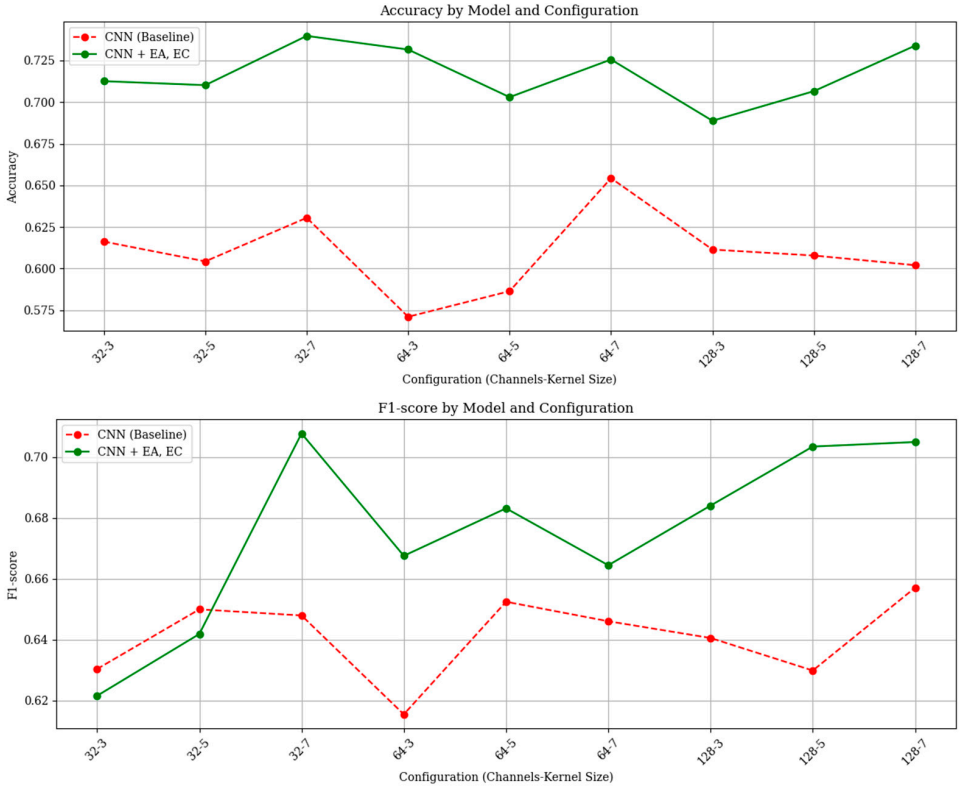


Figure 4. Comparison of results between the basic CNN model and the CNN model with EA and EC layers using the Receipt dataset.

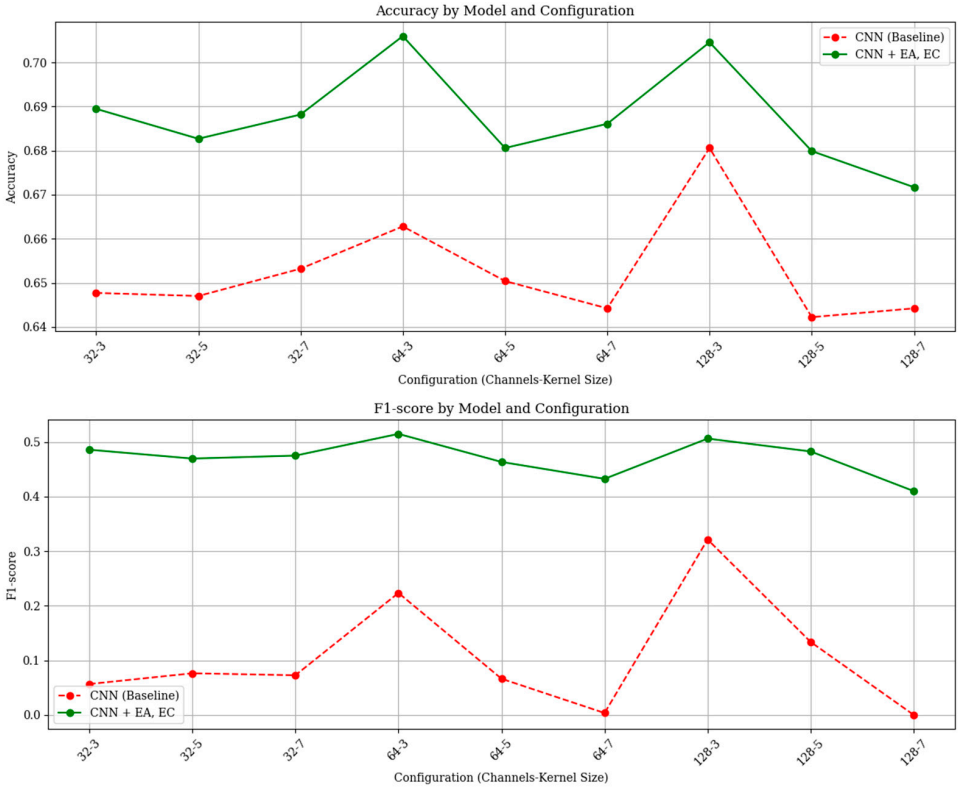


Figure 5. Comparison of results between the basic CNN model and the CNN model with EA and EC layers using the DocTamper dataset.

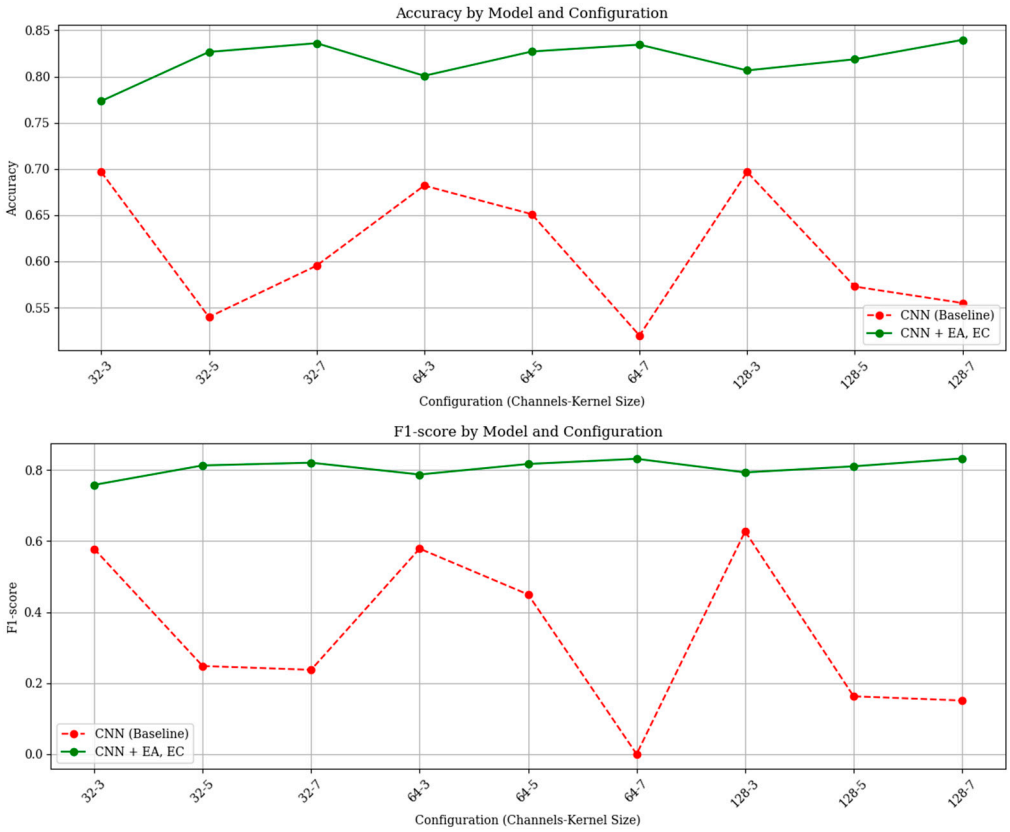


Figure 6. Comparison of results between the basic CNN model and the CNN model with EA and EC layers using the MIDV-2020 dataset.

As shown in Figures 4–6, the CNN model with added EA and EC layers generally achieved higher accuracy and F1-scores compared to the basic CNN model. This effect can be explained by the fact that in the basic CNN model, edge features may be underemphasized due to the presence of competing visual cues, which reduces the model's sensitivity to forged text. In contrast, the model with EA and EC layers explicitly enhances edge regions around the text and passes this focused information to the subsequent layers, thereby improving the model's performance to detect forged text.

4.4. Latency Evaluation for Practical Deployment

To evaluate the computational overhead introduced by the EA and EC layers, we measured the inference latency of the baseline CNN and the baseline model augmented with these layers across three datasets. The latency was measured for a single 128 × 128 image using real-time inference timing with Python 3.11.13's time module. As shown in Table 6, the latency increased significantly when the EA and EC modules were applied. For instance, the latency rose from 261.48 ms to 544.91 ms in the MIDV-2020 dataset. Similar increases were observed in the DocTamper and Receipt datasets. This increase is primarily due to the Sobel-based edge extraction operations and additional computations introduced by the EA and EC layers. Although this increase in latency may be a concern for real-time applications, the experimental results in Section 4.3 demonstrate that the inclusion of the EA and EC layers leads to substantial improvements in detection accuracy and F1-score. This trade-off between computational cost and performance suggests that the proposed modules are well-suited for applications where detection accuracy is prioritized over inference speed.

Table 6. Inference latency of the baseline model and the baseline model with EA and EC layers on three datasets.

Dataset	CNN	CNN + EA, EC Layers
Doctamper	264.40 ms	550.51 ms
Receipt	334.87 ms	555.73 ms
MIDV-2020	261.48 ms	544.91 ms

4.5. Ablation Study of EA and EC Layers Using

In this section, an ablation study was conducted on the EA and EC layers using not the basic CNN model, but well-established image classification and document forgery detection models whose performance has been validated in previous studies. The models used in the experiments include the image classification models DenseNet121, ResNet50, and Vision Transformer (ViT), as well as the document forgery detection model CAE-SVM proposed in [15]. For DenseNet121, ResNet50, and Vision Transformer (ViT-base with patch size 16 and input resolution 224), we adopted versions pre-trained on the ImageNet dataset to leverage robust feature representations. All layers in the backbone networks were fine-tuned, allowing the networks to adapt fully to the document forgery detection task during training. The datasets used for the experiments were the DocTamper, Receipt, and MIDV-2020 datasets. For each model, performance was first evaluated without the EA and EC layers by inputting original and forged text images. Then, the EA and EC layers were incorporated into the models, and performance was evaluated using the same procedure for comparison. In the case of DenseNet121, ResNet50, and ViT, the EA and EC layers were placed immediately after the input layer so that the enhanced edge features could be propagated through the entire model. For the CAE-SVM model, the EA and EC layers were inserted within the encoder to ensure that the edge-enhanced features were captured before down-sampling and effectively delivered to the decoder. Each model was evaluated in terms of classification accuracy and F1-score using five-fold cross-validation, and the results are presented in Table 7.

Table 7. Classification accuracy and F1-score of four models with and without EA and EC layers on three document forgery detection datasets.

Model	Type	DocTamper		Receipt		MIDV-2020	
		ACC	F1	ACC	F1	ACC	F1
DenseNet121	Base Model	0.49	0.66	0.57	0.70	0.54	0.0938
	Base Model + EA, EC layers	0.73	0.67	0.73	0.78	0.59	0.70
ResNet50	Base Model	0.59	0.60	0.54	0.35	0.58	0.21
	Base Model + EA, EC layers	0.69	0.71	0.61	0.62	0.70	0.55
CAE-SVM	Base Model	0.64	0.64	0.77	0.76	0.90	0.90
	Base Model + EA, EC layers	0.72	0.72	0.83	0.83	0.95	0.95
ViT	Base Model	0.70	0.69	0.58	0.70	0.83	0.85
	Base Model + EA, EC layers	0.72	0.68	0.80	0.82	0.93	0.93

The integration of the EA and EC layers consistently led to improved performance across all four models in the document forgery detection task. This improvement can be attributed to the fact that these layers effectively emphasize edge information within document images, thereby enabling the models to better learn subtle visual cues such as boundary inconsistencies and structural discontinuities introduced by manipulation. In the

case of DenseNet121, the accuracy on the DocTamper dataset increased significantly from 0.49 to 0.73, while maintaining a relatively high F1-score, indicating that the proposed layers contribute to performance gains even in high-performing models. ResNet50, on the other hand, showed a notable improvement on the Receipt dataset, with the F1-score increasing from 0.35 to 0.62. This suggests that the EA and EC layers can also enhance the sensitivity to boundary-level manipulations in models with initially lower baseline performance. Furthermore, in the document forgery-specific model CAE-SVM, the addition of the EA and EC layers improved all evaluation metrics. On the Receipt dataset, the accuracy increased from 0.77 to 0.83, and the F1-score from 0.76 to 0.83. Additionally, the Vision Transformer (ViT) model, which is known for its global context modeling capabilities, also benefited from the integration of the EA and EC layers. Notably, on the MIDV-2020 dataset, the accuracy increased from 0.83 to 0.93, and the F1-score from 0.85 to 0.93, demonstrating enhanced localization of manipulated regions even in transformer-based architectures. These results demonstrate that the proposed EA and EC layers are effective not only in general-purpose image classification models but also in specialized forgery detection frameworks, as they enable more precise localization and differentiation of forged regions by focusing on edge-specific features.

5. Conclusions

In this paper, an edge-enhancement method designed for integration into deep learning-based detectors has been proposed to improve the detection of manipulated document images. Unlike natural images, document images contain critical visual information that is highly concentrated along text boundaries. Manipulations often introduce distortions in these edge regions, which could serve as valuable cues for forgery detection. To address the limitations of conventional CNN-based models in capturing such subtle edge features, we developed two lightweight and modular components: the Edge Attention (EA) layer and the Edge Concatenation (EC) layer. These layers leverage Sobel-based edge extraction to emphasize edge-responsive channels and directly inject edge features into the network's representation space.

The proposed method was designed for compatibility with various model architectures, and its effectiveness was evaluated through extensive experiments across multiple benchmark datasets. Notably, consistent performance improvements were observed in both lightweight and complex models, including DenseNet121, ResNet50, ViT, and CAE-SVM. These experimental results demonstrated the scalability and generalizability of the proposed approach in diverse document forgery detection scenarios. By explicitly incorporating edge-specific features into the learning process, our method enhanced a model's ability to detect fine-grained structural inconsistencies—hallmarks of forged documents.

Beyond the technical improvements, the proposed method contributed to a broader understanding of how local structural features—especially edges—could be leveraged to improve document image forensics. The findings could support a design paradigm in which interpretable, edge-aware modules could be flexibly integrated into a variety of deep learning architectures without significant computational or architectural overhead. This has important implications for applications such as digital forensics and automated document verification, where transparency and robustness are critical.

Nevertheless, some limitations remain. The inclusion of EA and EC layers increases inference latency, which may be a drawback for real-time systems. Moreover, while the modules were tested on a range of representative architectures, they were not exhaustively evaluated across all types of forgery detection models. Finally, the method may be less effective in cases where forgery does not significantly alter edge structures, such as semantic-

level modifications or subtle color changes. These limitations point to future opportunities for enhancing the generality and efficiency of edge-aware detection systems.

Author Contributions: Conceptualization, Y.-Y.B. and K.-H.J.; validation, K.-H.J. and D.-J.C.; formal analysis, Y.-Y.B. and K.-H.J.; writing—original draft preparation, Y.-Y.B. and K.-H.J.; writing—review and editing, D.-J.C. and K.-H.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2021R1I1A3049788).

Data Availability Statement: The datasets used in this paper are publicly available and can be accessed through the following links: DocTamper dataset: https://github.com/qcf-568/DocTamper (accessed date 3 July 2024). Receipt dataset: http://l3i-share.univ-lr.fr/2023Finditagain/findit2.zip (accessed date 6 June 2024). MIDV-2020 dataset: http://l3i-share.univ-lr.fr/MIDV2020/midv2020.html (accessed date 25 June 2025).

Acknowledgments: We thank the anonymous reviewers for their valuable suggestions that improved the quality of this article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Bae, Y.-Y.; Cho, D.-J.; Jung, K.-H. A New Log-Transform Histogram Equalization Technique for Deep Learning-Based Document Forgery Detection. *Symmetry* **2025**, *17*, 395. [CrossRef]
2. Sharma, P.; Kumar, M.; Sharma, H. Comprehensive Analyses of Image Forgery Detection Methods from Traditional to Deep Learning Approaches: An Evaluation. *Multimed. Tools Appl.* **2023**, *82*, 18117–18150. [CrossRef]
3. Mahfoudi, G.; Morain-Nicolier, F.; Retraint, F.; Pic, M.M. CMID: A New Dataset for Copy-Move Forgeries on ID Documents. In Proceedings of the 2021 IEEE International Conference on Image Processing (ICIP), Anchorage, AK, USA, 19–22 September 2021; pp. 3028–3032.
4. Li, L.; Bai, Y.; Zhang, S.; Emam, M. Document Forgery Detection Based on Spatial-Frequency and Multi-Scale Feature Network. *J. Vis. Commun. Image Represent.* **2025**, *107*, 104393. [CrossRef]
5. Sun, F.; Sun, L.; Wang, J. Document Image Forgery Detection Based on Multiscale Edge Similarity. In Proceedings of the International Conference on Image Processing and Artificial Intelligence (ICIPAI 2024), SPIE, Online, 19 July 2024; Volume 13213, p. 132132S.
6. Qu, C.; Liu, J.; Zhang, H.; Chen, Y.; Wu, Y.; Li, X.; Yang, F. Robust Document Image Forgery Localization Against Image Blending. In Proceedings of the 2022 IEEE 21st International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Wuhan, China, 28–30 October 2022; pp. 810–817.
7. Zanardelli, M.; Guerrini, F.; Leonardi, R.; Adami, N. Image Forgery Detection: A Survey of Recent Deep-Learning Approaches. *Multimed. Tools Appl.* **2023**, *82*, 17521–17566. [CrossRef]
8. Gornale, S.S.; Patil, G.; Benne, R. Document Image Forgery Detection Using RGB Color Channel. *Trans. Eng. Comput. Sci.* **2022**, *10*, 1–14. [CrossRef]
9. Fridrich, J.; Soukal, D.; Lukás, J. Detection of Copy-Move Forgery in Digital Images. *Int. J. Comput. Sci. Issues* **2003**, *3*, 55–61.
10. Popescu, A.C.; Farid, H. Exposing Digital Forgeries by Detecting Duplicated Image Regions. In Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents VI, San Jose, CA, USA, 20–24 January 2004; Volume 5020, pp. 55–65.
11. Amerini, I.; Ballan, L.; Caldelli, R.; Del Bimbo, A.; Serra, G. A SIFT-Based Forensic Method for Copy–Move Attack Detection and Transformation Recovery. *IEEE Trans. Inf. Forensics Secur.* **2011**, *6*, 1099–1110. [CrossRef]
12. Christlein, V.; Riess, C.; Jordan, J.; Riess, C.; Angelopoulou, E. An Evaluation of Popular Copy-Move Forgery Detection Approaches. *IEEE Trans. Inf. Forensics Secur.* **2012**, *7*, 1841–1854. [CrossRef]
13. Pan, X.; Zhang, X.; Lyu, S. Exposing Image Splicing with Inconsistent Local Noise Variances. In Proceedings of the 2012 IEEE International Conference on Computational Photography (ICCP), Seattle, WA, USA, 13–15 May 2012; pp. 1–10.
14. Nandanwar, L.; Lu, Y.; Vincent, N.; Yuen, P.C.; Zheng, W.S.; Cheriet, F.; Suen, C.Y. A New Method for Detecting Altered Text in Document Images. In Proceedings of the International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI 2020), Chambéry, France, 1–5 June 2020; Lecture Notes in Computer Science. Springer: Cham, Switzerland, 2020; Volume 12068, pp. 96–107.

15. Tyagi, P.; Agarwal, K.; Jaiswal, G.; Sharma, A.; Rani, R. Forged Document Detection and Writer Identification Through Unsupervised Deep Learning Approach. *Multimed. Tools Appl.* **2024**, *83*, 18459–18478. [CrossRef]
16. Zhang, J.-Y.; Chen, Y.; Huang, X.-X. Edge Detection of Images Based on Improved Sobel Operator and Genetic Algorithms. In Proceedings of the 2009 International Conference on Image Analysis and Signal Processing (IASP), Linhai, China, 11–12 April 2009; pp. 31–35.
17. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2261–2269.
18. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
19. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image Is Worth 16×16 Words: Transformers for Image Recognition at Scale. In Proceedings of the 9th International Conference on Learning Representations (ICLR), Virtual Event, 3–7 May 2021.
20. Macit, H.B.; Koyun, A. An Active Image Forgery Detection Approach Based on Edge Detection. *Comput. Mater. Continua* **2023**, *75*, 1603–1619. [CrossRef]
21. Zafar, A.; Aamir, M.; Mohd Nawi, N.; Arshad, A.; Riaz, S.; Alruban, A.; Dutta, A.K.; Almotairi, S. A Comparison of Pooling Methods for Convolutional Neural Networks. *Appl. Sci.* **2022**, *12*, 8643. [CrossRef]
22. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
23. Sun, K.; Xiao, B.; Liu, D.; Wang, J. Deep High-Resolution Representation Learning for Human Pose Estimation. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 5693–5703.
24. Qu, C.; Liu, J.; Zhang, H.; Chen, Y.; Wu, Y.; Li, X.; Yang, F. Towards Robust Tampered Text Detection in Document Image: New Dataset and New Solution. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 18–22 June 2023; pp. 5937–5946.
25. Tornés, B.M.; Taburet, T.; Boros, E.; Rouis, K.; Doucet, A.; Gomez-Krämer, P.; Sidere, N.; d'Andecy, V.P. Receipt Dataset for Document Forgery Detection. In Proceedings of the 2023 17th International Conference on Document Analysis and Recognition (ICDAR), San José, CA, USA, 21–26 August 2023; pp. 454–469.
26. Bulatov, K.B.; Emelianova, E.V.; Tropin, D.V.; Skoryukina, N.S.; Chernyshova, Y.S.; Sheshkus, A.V.; Usilin, S.A.; Ming, Z.; Burie, J.-C.; Luqman, M.M.; et al. MIDV-2020: A Comprehensive Benchmark Dataset for Identity Document Analysis. In Proceedings of the 2022 Conference on Cybernetics and Education (KO), Moscow, Russia, 29 June–1 July 2022; pp. 1–10.