A W-Structured Encoder-Decoder Network Combining Swin Transformer and CNN for Image Inpainting Forensics

Mohamed Fathy Mohamed, Hala Abdel-Galil ElSayed, and Soha Ahmed Ehssan Aly

***Abstract*—Image inpainting is rapidly developing currently due to the progress of deep learning techniques and generative models. This has led to a loss of integrity in digital content, where inpainting techniques enable the production of highly realistic altered images that cannot be easily detected. To address these risks, this paper proposes a new deep learning–based image inpainting forensic network called W2SC-Net. The proposed architecture adopts a W-structured encoder-decoder design that integrates the Swin transformers with convolutional neural networks (CNNs). Specifically, the encoder block consists of two parallel streams to effectively extract both local textures and global contextual information. The decoder block is connected with the downsampling stages to enable accurate reconstruction. In addition, a high-pass filtered enhancement block is employed to highlight inpainting artifacts. Extensive experiments demonstrate not only the high detection performance of the proposed model but also its strong generalization capability. Although it was trained only on one inpainting method, it can accurately detect image manipulations across ten inpainting methods and diverse image datasets. Moreover, the W2SC-Net's robustness against anti-forensics attacks is further improved by introducing an additional training process. Finally, the W2SC-Net outperforms state-of-the-art forensic approaches in terms of F1-score and AUC evaluation metrics.**

***Index Terms*—deep learning, image inpainting forensics, inpainting detection, convolutional neural network CNN, Swin transformer, dual-stream feature extraction.**

I. Introduction

IMAGE inpainting is the automatic process of filling in visually realistic content to missing parts of an image by using the image's overall structure and contextual information surrounding the target area. Image inpainting techniques have been developed in two major stages: traditional techniques, which are further divided into patch inpainting [1]–[3] and diffusion inpainting [4], [5], and deep learning–based techniques [6]–[11]. Due to these techniques' efficient editing capabilities, image inpainting has gained widespread applications across many image processing fields [12]. Some of these applications are distorted image restoration, removing unwanted objects, medical image enhancement, satellite image completion, etc. On the other hand, these same capabilities have the potential to become a significant security risk regarding information security when used inappropriately. The situation becomes even worse when deep learning–based inpainting methods have become prevalent, as a malicious attacker can very easily change the facial content or erase the key objects/watermarks to be used in court as evidence or report fake news [13].

The opposite operation of image inpainting has emerged as a significant area of research to respond to these risks, which is called image inpainting detection or inpainting forensics. This field seeks to build a robust model that can identify the inpainted regions of an image, whether classifying the full image as forged/pristine or outputting a binary segmentation mask that represents the manipulated pixels. Paper [14] divides the forgery detection methods into two main categories: active and passive. Despite both types having the same goal of detecting forgery, they are different in the verification procedure. The active methods focus on embedding some metadata onto the images at the time of creation, and it can be later used to validate the authenticity of the image. The passive (or blind) methods do not offer so much specific information; thus, one has to rely entirely on the possible artifacts introduced in the tampering process. Because the passive has wider applicability in real-world scenarios than the active, it has gained increasing attention in recent years, and it is our focus in this research paper.

Early inpainting forensic techniques focused on detecting contextual inconsistencies or traces by relying on hand-crafted features and statistical analysis, such as [15]–[18]. Then, deep learning techniques are used in recent inpainting forensic approaches to address the limitations of hand-crafted feature extraction by learning features automatically. Some studies [19]–[28] designed models to detect a specific type of inpainting, whereas others [13], [29]–[34] proposed generalized frameworks capable of identifying any type of inpainting. Even though progress has been made in the field of inpainting forensics, its number of studies is still relatively few compared to the speed of inpainting techniques. Furthermore, there are still some issues and challenges that need to be addressed with respect to the state-of-the-art forensic methods. These issues can be summarized as (1) limitation of extracting global inpainting artifacts due to CNN-restricted receptive fields, (2) high computational and memory costs, (3) weak generalizabil-

Manuscript received November 19, 2025; revised December 9, 2025. Date of publication March 31, 2026. Date of current version March 31, 2026.

Authors are with the Computer Science Department, Faculty of Computers and Artificial Intelligence, Helwan University, Cairo, Egypt (e-mails: mfathy@fci.luxor.edu.eg, hala.nagy@fci.helwan.edu.eg, dr.soha@fci.helwan.edu.eg). M. F. Mohamed is also with the Computer Science Department, Faculty of Computers and Information, Luxor University, Luxor, Egypt.

Digital Object Identifier (DOI): 10.24138/jcomss-2025-0247

ity to unknown or various types of inpainting methods, and (4) poor detection performance under anti-forensics attacks.

In this paper, we address the aforementioned limitations and challenges by proposing a novel end-to-end deep learning–based inpainting detection model called W2SC-Net. W2SC-Net is a generalized inpainting forensic framework capable of identifying inpainted regions across both traditional and deep learning–based inpainting techniques. Our main contributions, and how the proposed W2SC-Net model addresses the outlined challenges, are summarized as follows:

1) A new image inpainting forensic model is proposed whose architecture efficiency can be summarized as: (1) leveraging the Swin transformer and CNN as dual encoder streams that not only extract features efficiently but also improve computational efficiency through parallel processing; (2) Swin-based decoder block that utilizes local and global extracted features for better reconstruction, and prevents the loss of image details through skip connections; (3) learnable high-pass filters are introduced as an enhancement block that allows the network to more focus on manipulation artifacts rather than irrelevant image content; and (4) combined loss function consisting of BCE and Dice losses is employed to improve the model's supervision during training and address the class imbalance problem. This architecture was carefully examined, and the results present its strong ability to detect subtle or complex inpainting manipulations.
2) The experimental results of the proposed model, which was trained on one inpainting method [9], present its strong generalizable detection capability to other inpainting methods. Among them are five traditional inpainting [1]–[5] and four deep learning inpainting [6], [7], [10], [11].
3) A quantitative comparison is conducted with several state-of-the-art forensic methods [13], [23], [29], [34], and the results demonstrate the superior detection performance of the proposed W2SC-Net model.
4) An enhanced version of the proposed W2SC-Net is also introduced, which achieves high detection performance under anti-forensic attacks.

The rest of this paper is organized as follows. Section II reviews the related work on inpainting forensics. Section III presents the proposed W2SC-Net model in detail. Experimental results are given in Section IV , and the conclusion is in Section V.

II. Related Work

Significant research efforts have been conducted in response to the information security risks posed by the malicious utilization of image inpainting. Based on the detection methodologies used, these studies can be classified into two major categories.

A. Traditional Feature-Based Forensics

Early forensic research predominantly focused on detecting pixels that were manipulated using patch-based inpainting techniques. In [15], a forgery detection algorithm is presented that integrates central pixel mapping, greatest zero-connectivity component labeling, and fragment splicing detection. In the [16] study, inpainted patches were detected via a hybrid feature of Euclidean, positional distance, and pixel overlap. For the detection of manipulated pixels in diffusion-based inpainted images. The method in [17] employs directional Laplacian analysis, multi-channel variance features, and specialized post-processing operations, while [18] introduced a detection method that applies weighted least squares filtering.

B. Deep Learning–Based Forensics

Several studies, such as [19]–[22], [28], have developed models for identifying pixels that were inpainted using traditional methods (patch-based or diffusion-based techniques). Paper [19] introduces the first CNN-based inpainting detection method. Authors in [20] proposed a hybrid CNN-LSTM forensic architecture with a filtering module. A 20-layer fully convolutional neural network architecture was proposed in [21]. The research paper [22] detects diffusion-based inpainting by integrating a feature pyramid network with an improved U-shaped net. For identifying any type of traditional-based inpainting, [28] utilizes a feature pyramid network with residual connections.

Other studies concentrate on identifying pixels that were manipulated using deep learning–based inpainting methods, like [23]–[27]. The study [23] employed a fully convolutional network based on high-pass-filtered image residuals, which enhanced the difference between the inpainted and untouched regions. In [24], a data generation approach is proposed to generate a training dataset that captures noise discrepancies. The study [25] combines concurrent spatial and channel attention. Paper [26] presents an attention-based feature pyramid network that integrates multi-scale convolution attention and fuses them with low-level features. A multi-path forensic network with a boundary guidance module is presented in [27].

To enable broad inpainting forensics, several approaches [13], [29]–[34] adopt a generalized framework that identifies manipulated pixels regardless of the inpainting technique (traditional or deep learning–based). The paper [30] proposes a detection method that integrates FPN with back connections. The paper [29] introduces an end-to-end deep neural network designed to detect various types of image forgeries, including inpainting. Authors in [13] proposed a detection network that was optimized using the NAS algorithm and integrates global-local attention mechanisms, and then this model was improved in [32] by introducing a near-original image augmentation technique. Paper [33] introduces an attention-based network that includes dual-stream frequency-spatial feature extraction. Transformer architectures have been used in some recent studies, such as [31], [34], to overcome the inherent drawbacks of CNN-based methods. In study [35], the authors enhanced the forensic architecture of [13] by combining an adaptive difference convolution block with a DenseNet block. Paper [36] introduced the first work that addresses inpainting prediction inaccuracies by considering both edge uncertainty and semantic inconsistency. Paper [37] proposed a dense feature

interaction network that employs a multi-scale feature pyramid architecture and adaptively combines low-level edge and shape features with high-level semantic features. The paper [38] proposes a forensic network that leverages features at different scales using a specially designed multi-scale extraction module and uses a reverse attention module as the backbone of the decoder.

Although notable progress has been made in inpainting forensics, there are several problems that still need to be solved to mitigate the danger of inpainting. Many state-of-the-art models fail to capture sufficient artifacts to detect inpainting in subtle, spatially dispersed, or structurally well-integrated cases. Complex models incur high computational and memory costs. Two critical aspects essential for a robust forensic model that are often overlooked are generalization and anti-forensics. To address these issues, we propose a deep learning–based generalized inpainting forensic framework called W2SC-Net.

III. Proposed Model: W-Structure Swin-CNN Network (W2SC-Net)

In this section, the W2SC-Net proposed model is presented in detail. W2SC-Net is a novel deep learning generalized framework specifically designed to detect and localize any type of inpainted pixels (diffusion-based, patch-based, or deep learning–based) within digital images. The term ”W2SC-Net” stands for W-Structure Swin-CNN Network, which employs an encoder-decoder architecture that resembles the shape of the letter ”W”. This architecture integrates convolutional neural networks (CNNs) with Swin transformers in a hybrid design to leverage the strengths of both local feature extraction and long-range contextual representation. An overview of the W2SC-Net architecture is illustrated in Figure 1.

The W2SC-Net generally consists of three key components: the enhancement block, the dual-stream encoder block, and the decoder block. The training and inference pipeline proceeds as follows: the input is a 3-channel RGB image of size (256×256), which first passes through the enhancement block to generate refined features. These enhanced features are then fed into two parallel encoder streams: the CNN encoder stream and the Swin encoder stream. The CNN stream consists of hierarchical convolutional stages, while the Swin stream is composed of hierarchical Swin transformer stages. The extracted features from both encoder streams are fused in the encoder bottleneck. The resulting high-level representation is then passed to the decoder block, which consists of a sequence of upsampling stages. Each upsampling stage integrates three sources of information: the main upsampled features, skip connections from the corresponding CNN encoder stage, and skip connections from the corresponding Swin transformer stage. The final output of the decoder block is a segmentation mask with the same spatial resolution as the input image, where each pixel represents a probability value in the range (0, 1), obtained via a Sigmoid activation function. A threshold (e.g., 0.5) can be applied during inference to obtain a binary mask. During training, a combined Dice and Binary Cross-Entropy (BCE) loss function is used to supervise the model by comparing the predicted mask with the ground truth, enabling accurate localization of inpainted regions.

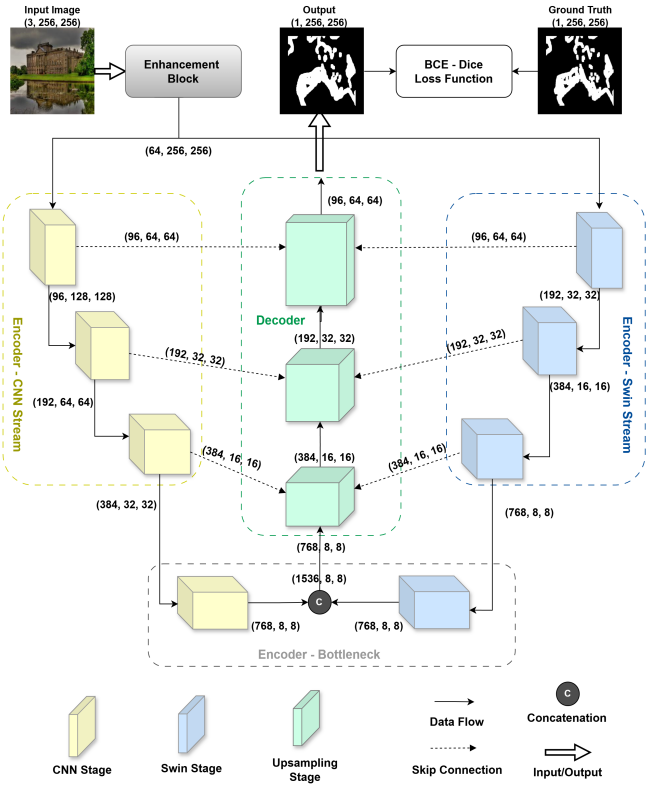
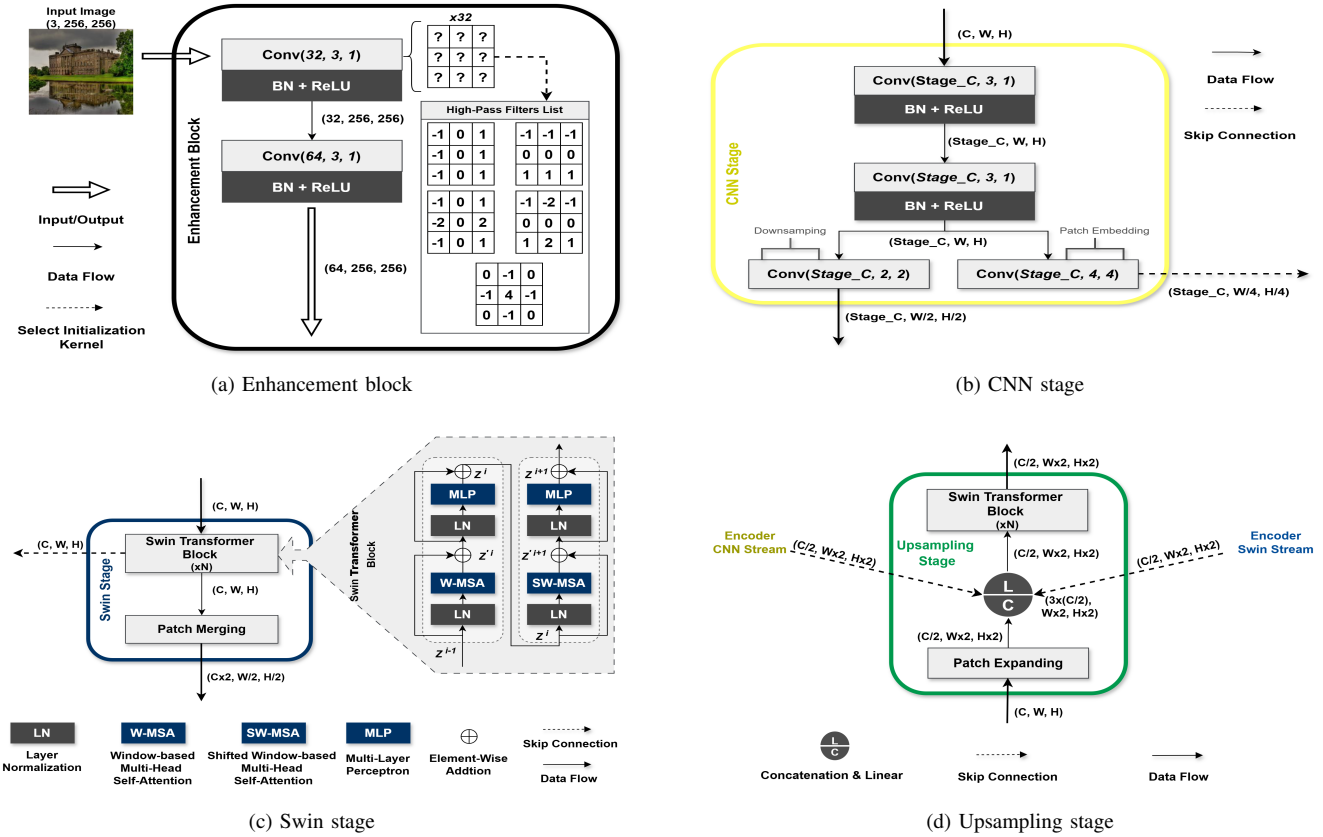


Fig. 1. The overview architecture of the W2SC-Net proposed model. The data flow is annotated with shapes in the format *(C, W, H)*, where *C* denotes the number of channels, *W* the width, and *H* the height of the feature map

A. Enhancement Block

The main goal of the enhancement block, which serves as the first stage in the W2SC-Net architecture, is to highlight inconsistencies in texture, structure, and frequency that commonly arise from inpainting operations. This helps suppress the semantic content of the input image and enhances the visibility of inpainting traces. Figure 2a illustrates the structure of the proposed enhancement block.

The enhancement block consists of two sequential convolutional blocks. Each block is composed of a convolutional layer (Conv), followed by batch normalization (BN) and a ReLU activation function. The first convolutional block expands the 3-channel input image by applying 32 learnable convolutional filters to highlight low-level residual cues commonly introduced by inpainting algorithms. The resulting feature maps are then further processed by the second convolutional block to produce the final enhanced output. This block doubles the capacity (to 64 learnable convolutional filters) to enrich the cues into more discriminative representations. The progressive channel expansion strategy is used in several forensic networks as an enhancement block, such as [13], [32], and [34]; therefore, it is adopted in the proposed model. We start the expansion with 32 channels to (1) capture diverse artifacts of inpainting, (2) maintain a moderate width to help the enhancement block focus on forensic residuals rather than semantic content, and (3) avoid abrupt dimensional jumps when transitioning into the encoder stages. This distribution



(a) Enhancement block (b) CNN stage

(c) Swin stage (d) Upsampling stage

Fig. 2. Key components of the W2SC-Net architecture. *Conv(f, k, s)* denotes a convolutional layer with f filters, a kernel size of $k \times k$, and a stride of s. Feature map shapes are annotated in the format *(Number of Channels, Width, Height)*

achieves the trade-off between effectively highlighting inpainting traces and keeping the enhancement block lightweight, as demonstrated by the enhancement block experiments in Section IV-F1. A common strategy for improving the detection of inpainting artifacts is to use high-pass filters to suppress the low-frequency components of the input image, which typically represent general content, while enhancing high-frequency details such as edges, textures, and artifacts. Inspired by this, we introduce a simple yet effective modification to the first convolutional block. The first convolutional layer is manually initialized using a set of predefined high-pass kernels instead of learning all weights from scratch. This strategy accelerates the learning process by biasing the network toward detecting fine structural details and anomalies from the very beginning of training. These initialized weights are then further optimized during training to better capture inpainting-specific patterns. The set of high-pass kernels used for initialization includes the Laplacian, Sobel, and Prewitt kernels, which are presented as follows:

- Laplacian Kernel:

$$K_{\text{Laplacian}} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \tag{1}$$

- Sobel Kernels:

$$K_{\text{Sobel-x}} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad K_{\text{Sobel-y}} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \tag{2}$$

- Prewitt Kernels:

$$K_{\text{Prewitt-x}} = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, \quad K_{\text{Prewitt-y}} = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \tag{3}$$

The 32 learnable filters are initialized based on the five kernels: $[K_{\text{Laplacian}}, K_{\text{Sobel-x}}, K_{\text{Sobel-y}}, K_{\text{Prewitt-x}}, K_{\text{Prewitt-y}}]$. So each kernel type is assigned to six filters, resulting in $5 \times 6 = 30$ filters. The remaining two are assigned to the first two elements in the list, namely $K_{\text{Laplacian}}$ and $K_{\text{Sobel-x}}$. Consequently, the final distribution consists of 7 Laplacian kernels, 13 Sobel kernels, and 12 Prewitt kernels. The second convolutional block consists of normal convolutional layers, whose 64 filter weights are randomly initialized.

B. Encoder Block

The encoder block in W2SC-Net is designed to extract hierarchical feature representations from the enhanced input image. It progressively transforms low-level features (e.g., edges, colors) into high-level semantic features (e.g., contextual anomalies, forensic artifacts), capturing both local details

and global structural information. We integrate CNN and the Swin transformer within the encoder block using a dual-stream architecture to leverage the strengths of both architectures. CNNs efficiently extract local features and textures, while the Swin transformer captures global dependencies. By utilizing self-attention mechanisms, the Swin blocks enable the model to understand complex patterns and contextual information across the entire image. The following subsections provide a detailed explanation of the dual-stream architecture and its integration via the encoder bottleneck block.

1) CNN Stream: As shown in Figure 1, the CNN stream takes the output of the enhancement block as an input and extracts local features through three hierarchical CNN stages. Each CNN stage receives one input feature map and produces two outputs: one passed to the next stage and another forwarded as a skip connection to the corresponding decoder block. All CNN stages share the same structure, as illustrated in Figure 2b. Each CNN stage doubles the number of input channels, except for the first stage, which expands the input to 96 channels. Specifically, the values of these channel expansions, denoted by the Stage_C variable in the figure, are 96, 192, and 384. These values correspond to the channel sequence of the parallel encoder stream. Each CNN stage extracts features using two sequential convolutional blocks, each consisting of a convolutional layer (Conv), batch normalization (BN), and a ReLU activation function. The resulting feature maps are then routed in two directions: (1) to the next stage via a downsampling layer and (2) to the skip connection via a patch embedding layer. The downsampling layer is mainly used to reduce spatial dimensions (height and width) by half, while the patch embedding layer applies a 4×4 convolution to project the features in a format compatible with the decoder.

2) Swin Transformer Stream: Similar to the CNN stream, the Swin stream also takes the enhanced feature map as an input. But first, it applies a patch embedding layer before performing feature extraction. This layer splits the input feature map into non-overlapping patches and projects each patch into a fixed-length embedding vector suitable for transformer processing. The patch embedding is implemented using a convolutional layer with a kernel size and stride of 4, as illustrated in the bottom right block in Figure 2b. Figure 1 shows that the Swin stream comprises three hierarchical Swin stages, all following the same structure. This structure is illustrated in detail in Figure 2c.

The Swin stage extracts global features from the input feature map using a series of Swin transformer blocks. Each Swin stage contains a specific number of blocks, denoted by the variable N in the figure, set to (2, 2, 6) for Swin Stages 1, 2, and 3, respectively. This distribution follows the hierarchical design principle of the tiny-sized Swin transformer [39]. The Swin transformer blocks are computed as follows:

$$Z'^{\,\mathrm{i}} = \text{W-MSA}(\text{LN}(Z^{i-1})) + Z^{i-1} \tag{4}$$

$$Z^{i} = \text{MLP}(\text{LN}(Z'^{\,\mathrm{i}})) + Z'^{\,\mathrm{i}} \tag{5}$$

$$Z'^{\,\mathrm{i+1}} = \text{SW-MSA}(\text{LN}(Z^{i})) + Z^{i} \tag{6}$$

$$Z^{i+1} = \text{MLP}(\text{LN}(Z'^{\,\mathrm{i+1}})) + Z'^{\,\mathrm{i+1}} \tag{7}$$

Each Swin transformer block applies window-based multi-head self-attention (W-MSA) to capture local dependencies within non-overlapping windows. Then, shifted window-based multi-head self-attention (SW-MSA) is performed to enable cross-window interactions and enhance global context modeling. Both attention modules are preceded by Layer Normalization (LN) and followed by a Multi-Layer Perceptron (MLP), with residual connections applied after each component. The output of the last Swin block of each sequence will be passed to both the decoder as a skip connection and processed using patch merging for the next stage. Patch merging is a downsampling block that consists of a linear projection layer followed by layer normalization, which is used to reduce the spatial resolution while increasing the number of channels.

3) Bottleneck Block: The bottleneck block is the final component of the W2SC encoder and is responsible for fusing the features extracted from both streams before passing them to the decoder. Its architecture is shown in the (Encoder–Bottleneck) block in Figure 1. The output from the CNN stream is processed using the same structure as the CNN stage, which is described in Figure 2b, but without a downsampling layer. The output from the Swin stream is also processed using the same structure as the Swin stage, which is described in Figure 2c, with two Swin transformer blocks and without a patch merging layer. These two extracted features are then concatenated along the channel dimension, and a linear layer is applied to reduce the number of channels to 768, making the fused feature map compatible with the decoder stream.

C. Decoder Block

The decoder block in the proposed W2SC-Net model is responsible for progressively reconstructing the predicted manipulation mask from the high-level features extracted by the encoder. As illustrated in Figure 1, the decoder begins by upsampling these features through three hierarchical upsampling stages. Each stage reconstructs fine spatial details by combining the upsampled features from the previous stage with two types of skip connections: one from the CNN stream that provides local structure details and another from the Swin stream that contributes global contextual information. More specifically, these upsampling stages perform the inverse operation of the encoder's Swin stream, using the same Swin transformer block sequences but replacing patch merging with patch expansion. In more detail, Figure 2d shows the structure of each upsampling stage. In contrast to patch merging, the patch expansion operation increases the height and width of the feature map while reducing the channel dimension. These expanded features are concatenated channel-wise with the two types of skip connections, and then a linear layer is used

to reduce the number of channels before passing them to the Swin transformer sequence. The architecture of the Swin transformer block is shown in Figure 2c. The number of Swin transformer blocks per stage, denoted by the variable N in the figure, is set to (6, 2, 2) from bottom to top, reversing the sequence of the encoder's Swin stream. The output of the last upsampling stage is further processed using a final patch expansion layer, which is similar to the one described in Figure 2d, with the key differences that it does not change the number of input channels and it expands the spatial dimensions (width and height) by a factor of 4 to match the resolution of the input inpainted image. Finally, this expanded feature map is then processed by a (1×1) head convolution layer to reduce the channel dimension to 1, resulting in the final prediction mask.

D. BCE-Dice Combined Loss Function

The primary objective of the proposed W2SC-Net model is to distinguish between manipulated (inpainted) and authentic regions, essentially a per-pixel binary classification task. Therefore, we employ the Binary Cross-Entropy (BCE) loss function to supervise W2SC-Net during its training process. The BCE loss is defined as:

$$\mathcal{L}_{\text{BCE}}(\mathbf{G},\mathbf{O}) = -\frac{1}{H\times W}\sum_{r=1}^{H}\sum_{c=1}^{W}\Big[\mathbf{G}_{r,c}\times\log(\mathbf{O}_{r,c}) + (1-\mathbf{G}_{r,c})\times\log(1-\mathbf{O}_{r,c})\Big] \tag{8}$$

where H and W are the height and width of the mask, $\mathbf{G}_{r,c}$ denotes the $(r,c)^{\text{th}}$ pixel in the ground truth mask, and $\mathbf{O}_{r,c}$ denotes the corresponding pixel in the W2SC-Net output mask.

The BCE loss treats all pixels equally, which can be problematic in inpainting detection tasks where manipulated regions are often small compared to pristine regions. This imbalance may cause the model to become biased toward predicting the majority (pristine) class, resulting in poor localization of subtle inpainting traces. To address this problem, we propose combining the BCE loss function with the Dice loss function. This combination helps handle the class imbalance by explicitly measuring the overlap between the model's output mask and the ground truth mask. The Dice loss is defined as follows:

$$\mathcal{L}_{\text{Dice}}(\mathbf{G},\mathbf{O}) = 1 - \frac{2\sum_{r=1}^{H}\sum_{c=1}^{W}\mathbf{G}_{r,c}\times\mathbf{O}_{r,c}+\epsilon}{\sum_{r=1}^{H}\sum_{c=1}^{W}\mathbf{G}_{r,c}+\sum_{r=1}^{H}\sum_{c=1}^{W}\mathbf{O}_{r,c}+\epsilon} \tag{9}$$

where ϵ is a small constant added for numerical stability. In summary, the BCE-Dice combined loss function used in the proposed W2SC-Net model is defined as:

$$\mathcal{L}_{\text{BCE-Dice}} = \mathcal{L}_{\text{BCE}}(\mathbf{G},\mathbf{O}) + \mathcal{L}_{\text{Dice}}(\mathbf{G},\mathbf{O}) \tag{10}$$

This fusion leverages the strengths of both losses: BCE focuses on pixel-wise accuracy, while Dice loss emphasizes region-level agreement. As a result, it enhances the W2SC-Net model's sensitivity to small or imbalanced manipulated areas, thereby improving its detection accuracy.

IV. Experimental Results and Analysis

A. Dataset

To ensure a fair comparison with state-of-the-art methods, the same training and testing datasets as in [13] were used, enabling direct evaluation of performance improvements under identical conditions. The training set consists of 48,000 inpainted images evenly sampled from the Places [40] and Dresden [41] datasets. All images were manipulated using the GC [9] inpainting method. Each image was inpainted based on its corresponding binary mask, which is irregular in shape and randomly selected from [8]. To improve the proposed model's robustness and prevent overfitting, random horizontal flipping and random 90° rotation were applied as data augmentation techniques during the training phase.

As for the testing set, the data variety was increased by the use of additional image datasets (CelebA [42], ImageNet [43]) and the utilization of different shapes of masks (irregular, rectangles, circles, ellipses, and polylines). The test images are manipulated using a variety of inpainting methods to properly test the generalizability of the forensic model. The evaluation process was conducted using ten different inpainting methods that are equally divided between traditional-based and deep learning–based approaches. The traditional ones are TE [5], NS [4], PM [2], LR [3], and SG [1], while the deep learning–based ones are GC [9], CA [6], SH [7], LB [10], and RN [11]. Each inpainting method is applied to 1,000 original images to construct the test dataset.

B. Implementation Details

The proposed W2SC-Net model was implemented in Python using the PyTorch framework. Training and testing were conducted on the Kaggle platform using a Linux-based environment with a 64-bit x86-64 processor architecture. The system was equipped with a single NVIDIA Tesla P100-PCIE-16GB GPU to enable accelerated computation. The input images have dimensions of 256×256×3 (height × width × channels) and are normalized using the mean and standard deviation of ImageNet [43]. The Adam optimizer [44] with default parameters and a batch size of 14 is adopted for training. The initial learning rate is set to 1e-4 and is reduced by half if the validation loss does not decrease for five consecutive epochs, continuing this process until convergence. The Swin blocks in W2SC-Net are initialized with the weights of the pretrained Swin on ImageNet-1K. The W2SC-Net model was trained until convergence while avoiding overfitting, which was achieved after 70 epochs. Two primary metrics were used to evaluate and compare the model's performance: (1) the F1-score (harmonic mean of precision and recall) to assess class-wise balance and (2) AUC-ROC (Area Under the Receiver Operating Characteristic Curve) to measure threshold-agnostic discriminative power.

C. Quantitative Results

To evaluate the performance of the W2SC-Net proposed model in inpainting forensics, a comprehensive quantitative

TABLE I
AUC (%) COMPARISON BETWEEN THE PROPOSED W2SC-NET MODEL AND BASELINE FORENSIC MODELS

Models	Test Dataset										
	Deep Learning Inpainting					Traditional Inpainting					Mean
	GC [9]	CA [6]	SH [7]	LB [10]	RN [11]	TE [5]	NS [4]	LR [3]	PM [2]	SG [1]	
ManTra-Net [29]	96.31	75.44	73.58	62.27	87.38	90.93	89.03	97.12	90.59	86.09	84.87
HP-FCN [23]	96.65	87.50	98.14	96.51	96.59	92.27	97.18	99.18	98.64	99.78	96.24
IID-Net [13]	96.77	95.39	99.67	99.80	99.71	96.12	97.65	99.79	99.54	99.94	98.44
CT-Net [34]	98.25	98.22	98.52	99.71	99.83	99.05	98.69	99.87	99.82	99.91	99.19
W2SC-Net (Proposed Model)	99.80	98.00	99.89	99.84	99.86	99.11	99.54	99.95	99.83	99.93	99.58

TABLE II
F1-SCORE (%) COMPARISON BETWEEN THE PROPOSED W2SC-NET MODEL AND BASELINE FORENSIC MODELS

Models	Test Dataset										
	Deep Learning Inpainting					Traditional Inpainting					Mean
	GC [9]	CA [6]	SH [7]	LB [10]	RN [11]	TE [5]	NS [4]	LR [3]	PM [2]	SG [1]	
ManTra-Net [29]	92.10	19.02	32.78	2.38	10.80	83.23	86.75	27.37	13.11	45.84	41.34
HP-FCN [23]	76.93	35.75	81.43	55.78	56.58	41.05	44.13	50.91	24.66	73.55	54.08
IID-Net [13]	83.61	81.46	94.13	96.14	94.41	82.47	85.27	87.28	75.74	94.78	87.53
CT-Net [34]	88.78	85.12	95.68	94.06	95.05	83.71	85.49	90.54	82.82	94.15	89.54
W2SC-Net (Proposed Model)	95.79	86.22	96.50	95.57	96.04	91.42	92.38	92.86	81.45	95.29	92.35

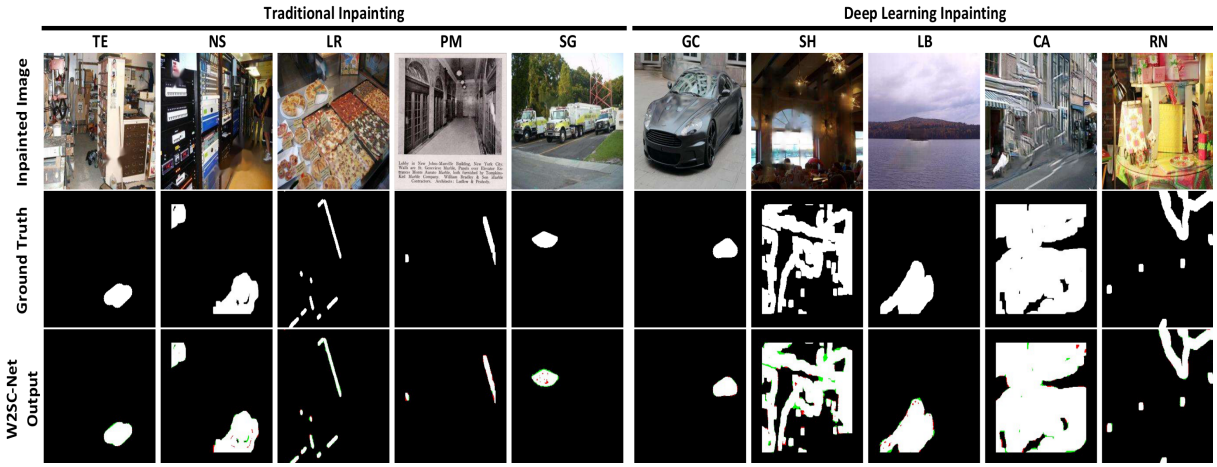
comparison is conducted against four existing state-of-the-art forensic methods: ManTra-Net [29], HP-FCN [23], IID-Net [13], and CT-Net [34]. All these approaches are deep learning–based models that were retrained and tested on the dataset from [13], which is the same dataset used for the proposed model. The detection results measured by AUC are presented in Table I, while the results measured by F1-score are presented in Table II. In both tables, the columns present the inpainting methods of the test dataset that are categorized based on the inpainting type. Thus, each row displays the performance of each forensic model evaluated across various inpainting methods. To evaluate the detection accuracy under consistent inpainting conditions, the gray columns in the tables highlight results for the GC [9], which is the same inpainting method used in training. To evaluate the generalization accuracy of the forensic models, the remaining inpainting method columns in the tables, from CA to SG, are used. The final column provides a comprehensive summary by displaying the mean AUC and F1-score for each model.

In terms of AUC, the proposed W2SC-Net model achieved the best performance in the case of training and testing using the same inpainting method, with 99.80% AUC (exceeding CT-Net's 98.25%). In generalizability evaluation, W2SC-Net achieved superior performance in most of the inpainting methods (7 out of 9 test cases), while ranking second in the remaining two (CA and SG), with small differences of 0.22 and 0.01, respectively. In general, the mean column indicates that all models performed relatively well in terms of AUC, with the lowest value being 84.87% for ManTra-Net. However, the proposed W2SC-Net model outperforms all baseline models, achieving 99.58% AUC and surpassing the second-best model, CT-Net, which recorded 99.19% AUC.

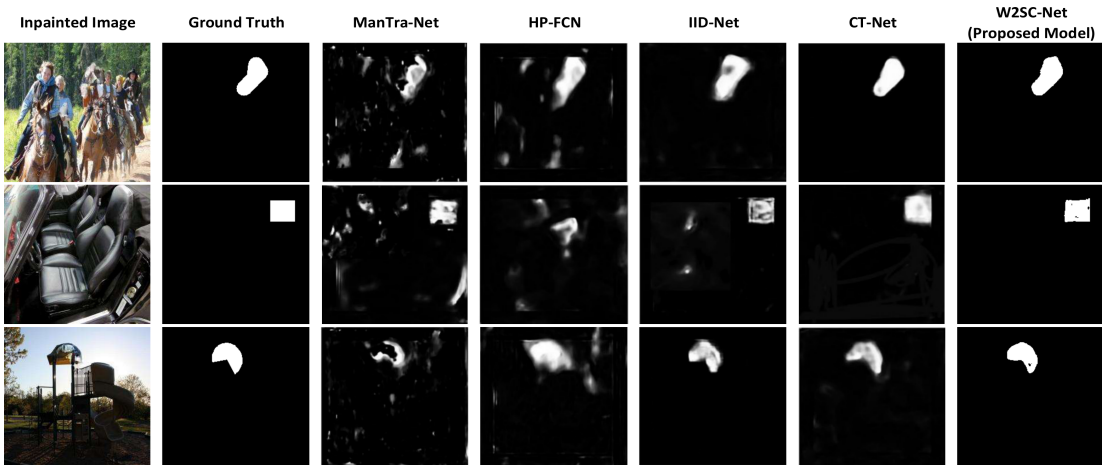
A similar trend is observed for the F1-score, where W2SC-Net again achieved the best results in the GC method with a 95.79% F1 (exceeding ManTra-Net's 92.10%) and superior performance in 7 out of 9 unseen methods. The two cases where it ranked second were the LB and PM methods, with differences of 0.57 and 1.37, respectively. Unlike AUC, the mean F1-score results show more significant disparities, where the two lowest values are 41.34% and 54.08% for ManTra-Net and HP-FCN, respectively. These differences highlight the limitations of some models in achieving a balanced precision-recall trade-off. However, the proposed W2SC-Net model addressed this problem and achieved the best results with a 92.35% F1, surpassing the second-best model, CT-Net, which recorded an 89.54% F1. These quantitative comparisons establish the state-of-the-art performance of the proposed W2SC-Net model in achieving both high inpainting detection accuracy and strong generalization capability across various techniques.

TABLE III
COMPUTATIONAL EFFICIENCY COMPARISON BETWEEN THE PROPOSED W2SC-NET AND BASELINE INPAINTING FORENSIC MODELS

Models	Parameters (million)	FLOPs (GFLOPs)	Inference Time (ms)	Peak GPU Memory (MB)
IID-Net [13]	5.78	73.56	43.01	668.04
CT-Net [34]	118.32	169.65	27.54	1261.21
Proposed Model	66.77	96.48	31.28	950.97



(a) Qualitative results of the proposed W2SC-Net model. In the "W2SC-Net Output" row, a white pixel indicates true positives, a black pixel indicates true negatives, a green pixel indicates false positives, and a red pixel indicates false negatives



(b) Qualitative comparison between the proposed W2SC-Net model and existing forensic models, ManTra-Net [29], HP-FCN [23], IID-Net [13], and CT-Net [34]

Fig. 3. Visual results of inpainting forensics

In addition to the inpainting detection performance comparison, the computational efficiency of the W2SC-Net proposed model is compared with recent approaches, IID-Net [13] and CT-Net [34], as shown in Table III. The provided information includes the number of trainable parameters, floating-point operations (FLOPs), inference latency, and peak GPU memory consumption, all of which were measured under the same hardware settings described in Section IV-B. IID-Net is the most lightweight model because it relies only on CNNs and attention mechanisms. Despite that, it shows comparatively slower inference performance. In contrast, CT-Net has the highest number of parameters and computational load due to its transformer-based architecture; however, it still achieves the fastest inference time. The proposed W2SC-Net model occupies a middle point across all metrics, where it requires substantially fewer parameters and memory than CT-Net and achieves faster inference speed than IID-Net. These results demonstrate that W2SC-Net achieves a balanced trade-off between detection accuracy, memory consumption, and processing time, which makes it suitable for practical forensic applications.

D. Qualitative Results

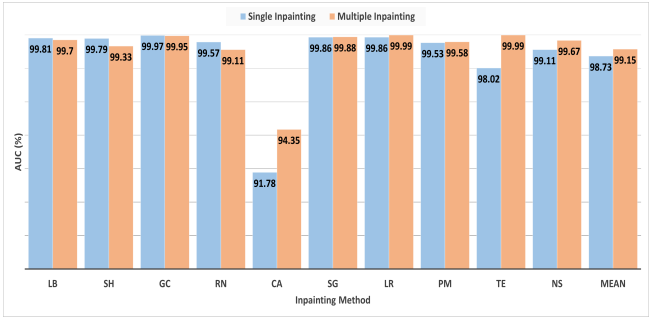
In this subsection, a comprehensive visual analysis is conducted to better validate the accuracy of the proposed W2SC-Net model in localizing inpainted regions. The results of these analyses, which include two types of experiments, are shown in Figure 3. Figure 3a presents the visual results of the proposed W2SC-Net model on detecting various inpainting techniques, including both types of methods (traditional and deep learning). Figure 3b presents a visual inpainting detection comparison between the proposed model and the state-of-the-art forensic models.

E. Analysis of Inpainting Detection

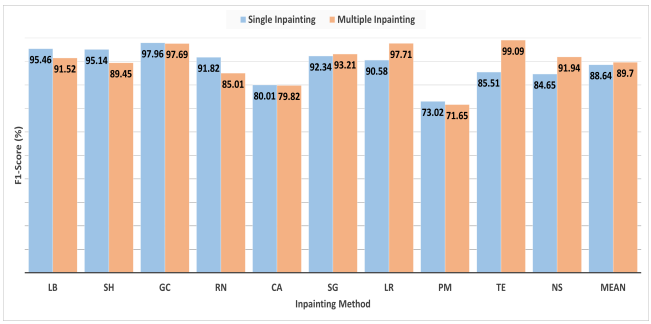
1) Single-Method vs. Multi-Method Inpainting Training: The experiments in [13] concluded that various inpainting methods leave common detectable traces. Therefore, the authors focused on improving the generalizability of the forensic model by training on a single inpainting method (GC [9]) and testing on several unseen methods. According to this strategy, the proposed W2SC-Net model achieved strong generalizability, as illustrated in Section IV-C. Another approach

that may improve the model's generalizability is to train the forensic model on a diverse but limited set of inpainting methods. This enables the model to recognize common traces across different types of inpainting, avoid overfitting to a specific method, and maintain evaluation on several unseen methods. This subsection presents a small experiment on the proposed W2SC-Net model to validate this theory. W2SC-Net is trained on two different datasets. The first dataset is manipulated using a single inpainting method (GC [9]). The second dataset is manipulated using three inpainting methods, each representing a different inpainting type (diffusion, patch, and deep learning), namely TE [5], LR [3], and GC [9]. Both datasets are initialized with 3,000 inpainted images sampled from the main training and testing sets. The dataset size is then increased to 15,000 images by applying simple data augmentation techniques, which include (1) horizontal flip, (2) vertical flip, (3) 90° rotation, and (4) random cropping. Figure 4 presents the results of this experiment using AUC and F1-score evaluation metrics. It can be observed that the multiple-inpainting version performs better in terms of the mean values of both evaluation metrics. However, when the three inpainting methods used for training are excluded to properly evaluate generalizability, both versions perform similarly. This behavior is likely because of the weakness of the training dataset, as the multiple-inpainting dataset requires a larger scale with actual manipulations to enable the model to effectively learn diverse inpainting patterns. As a result, developing a robust dataset that includes different inpainting methods in the training set could be a solution to further improve the model's generalizability in future work.

2) Manipulation Ratio Analysis: Another aspect of forensic analysis addressed in this subsection is how detection performance is influenced by the change of manipulation ratios. An experiment was conducted to measure the performance of the proposed W2SC-Net model at different levels of mask sizes, starting from 1% manipulation ratio to 60% manipulation ratio. The final (manipulation ratio → F1-score) results can be summarized as follows: (10 → 97.63), (20 → 98.49), (30 → 98.89), (40 → 98.2), (50 → 99.24), (60 → 99.44). It can be observed that there is a clear relationship between the inpainting rate and detection performance: as the manipulation ratio increases, the F1 score generally improves.



(a) AUC results



(b) F1-score results

Fig. 4. Effect of single-method versus multi-method inpainting training on W2SC-Net performance

F. Ablation Studies

Comprehensive studies were conducted to evaluate the contribution of key components in the W2SC-Net proposed forensic model. By systematically removing or modifying individual units, we assess their impact on detection performance and validate the effectiveness of our design choices. All ablation study training experiments follow the same configuration described in the Section IV-B, except for two changes: the number of training epochs is fixed at 20, and the training set is reduced to 20,000 inpainted samples. The details of these ablation experiments are presented in the following subsections.

1) Importance Analysis of W2SC-Net Components: To assess the contribution of each key component in the W2SC model, this study compares the performance of the full model against its ablated variants, where one component (enhancement block, encoder CNN stream, or encoder Swin stream) is removed at a time. Table IV presents the detection results of this study, which highlight three key findings: (1) the complete model achieves superior performance (97.82 AUC, 89.01 F1); (2) the relative importance of each component, as evidenced by performance degradation patterns; and (3) the number of parameters of each configuration. Notably, disabling the enhancement block results in the largest F1 score reduction (6.81 ↓), disabling the encoder CNN stream results in the largest AUC score reduction (0.46 ↓), and the encoder Swin stream accounts for the largest share of trainable parameters.

TABLE IV
IMPACT OF COMPONENT REMOVAL ON W2SC-NET PERFORMANCE. ENABLED COMPONENTS ARE MARKED WITH (✓), WHILE (✗) DISABLED ONES

W2SC Components			Number of Trainable Parameters	Overall AUC	Overall F1
Enhancement Block	Encoder CNN Stream	Encoder Swin Stream			
✓	✓	✓	66,770,437	97.82	89.01
✗	✓	✓	66,604,549	97.46	82.2
✓	✗	✓	41,491,525	97.36	86.43
✓	✓	✗	37,784,971	97.41	86.69

2) Transformer Selection for Inpainting Detection: In addition to the theoretical justification for the reason for selecting the Swin transformer in the proposed model, this study provides practical evidence by comparing the inpainting detection performance of three transformer-based architectures. The

TABLE V
IMPACT OF COMPONENT SUBSTITUTION ON W2SC-NET PERFORMANCE

Enhancement Block	Normal Conv	✓					✓	✓					
	High-Pass Conv		✓	✓	✓	✓	✓		✓	✓	✓	✓	✓
	Bayar			✓		✓	✓	✓					
	PF				✓	✓	✓	✓					
Decoder Block	Swin	✓	✓	✓	✓	✓	✓	✓			✓	✓	✓
	CNN								✓	✓			
Encoder Block	CNN	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓
	Swin	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Loss Function	BCE	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
	Dice	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	
	Focal												✓
	Focal Tversky												✓
Number of Trainable Parameters ($\times 10^4$)		6677	6677	6677	6677	6677	6677	6677	5838	3310	6677	6677	6677
Overall AUC		97.58	97.82	97.22	97.64	97.24	97.91	97.33	97.43	96.73	97.59	94.25	97.42
Overall F1		83.27	89.01	83.85	87.43	84.79	87.57	85.46	87.88	78.4	87.92	85.23	86.52

three architectures are the Vision Transformer (ViT) [45], the Swin Transformer [39], and the Masked-Attention Mask Transformer (Mask2Former) [46]. All models were trained for 20 epochs, and the training and validation accuracy values were computed using the AUC metric after each epoch. The models demonstrated similar learning patterns, though with notable performance differences. The training accuracy values were 88.58% for ViT, 99.61% for Swin, and 97.23% for Mask2Former. The validation accuracy values were 88.24% for ViT, 99.65% for Swin, and 97.29% for Mask2Former. As a result, the Swin Transformer emerged as the optimal choice for the proposed model.

3) Component-Wise Substitution Analysis of W2SC-Net: This subsection shows a systematic component replacement analysis for the W2SC-Net architecture, where each core module is replaced with its possible alternatives. Through comprehensive ablation experiments, we determine the best-specific configuration for image inpainting forensic tasks and how architectural variants perform relative to each other. These comprehensive ablation experiments are summarized in Table V. The experiments consist of 12 configuration variants, each represented by a column in the table. The substitutions are categorized into three main stages: enhancement block alternatives, encoder-decoder alternatives, and loss function alternatives. In the table, the first seven columns correspond to the enhancement block alternatives, the next two columns represent the encoder-decoder alternatives, and the final three columns are dedicated to the loss function alternatives. As described in the Section III, the enhancement block consists of two sequential convolutional blocks. In these experiments, we replace the first block with one or a combination of alternative candidates, while keeping the second block fixed. In addition to the normal convolution and high-pass initialized convolution (used in our proposed model), Bayer layer [47] and Pre-Filtering (PF) layer [23] are also considered as alternative candidates for the enhancement block. The first seven columns of the table illustrate most of the possible combinations of these four alternatives. When integrating multiple types of enhancement layers, each layer processes the input image independently, and their outputs are concatenated before being passed to the next enhancement block. In the encoder-decoder stage, CNN upsampling layers are employed in the decoder block instead of Swin upsampling. This modification is tested using two different encoder configurations: the Swin stream alone and a combination of CNN and Swin streams. To empirically validate the effectiveness of the proposed composite loss function in our model, we compare it against its components individually, BCE loss and Dice loss, as well as against another composite loss function that combines Focal loss [48] and Focal Tversky loss [49].

The results in the table show that the original W2SC-Net configuration achieves the best overall performance, with an Overall AUC of 97.82% and an Overall F1 score of 89.01%. Although the variant replacing the enhancement block with (Normal Conv, High-Pass Conv, Bayar, and PF) attains a slightly higher AUC (97.91%), its F1 score drops substantially (87.57%), a decrease of 1.44 points compared to the original. This trade-off confirms that the fully original W2SC-Net setup offers the optimal balance between discriminative power and practical detection effectiveness.

G. Robustness Evaluation against Anti-Forensic Operations

In real-world scenarios, post-processing operations—referred to as anti-forensics—are often applied after filling the missing regions using inpainting techniques. The goal of these operations is to make the inpainted areas appear visually seamless and statistically consistent with the surrounding image, thereby concealing traces of inpainting and evading forensic detection. A series of experiments was conducted to evaluate the robustness of the proposed W2SC-Net model under these challenging conditions. The experiments included five commonly used anti-forensic operations: image resizing, JPEG compression, additive white Gaussian noise (AWGN), Gaussian blur, and global histogram equalization. Each operation was applied with varying magnitudes to the test dataset. Specifically, the scaling

TABLE VI
AUC (%) DETECTION PERFORMANCE OF THE PROPOSED W2SC-NET MODEL AND ITS ENHANCED VERSION UNDER ANTI-FORENSIC ATTACKS

Proposed Model Versions	Without Anti-forensics	Resizing			Gaussian Noise ($\times 10^{-3}$)			JPEG Compression			Gaussian Blur			Histogram Equalization
		95%	90%	85%	3	6	9	95%	90%	85%	0.5	0.7	1	
W2SC-Net	98.87	94.97	88.8	85.57	98.77	96.47	84.93	97.45	97.98	88.69	98.93	90.16	72.34	93.54
Enhanced W2SC-Net	99.41	99.04	99.22	99.01	99.41	99.43	99.05	99.13	99.07	98.96	99.39	99.17	98.02	94.72

factor was varied for resizing, the standard deviation was varied for Gaussian noise and Gaussian blur, and the quality factor was varied for JPEG compression. Table VI reports the overall AUC detection performance of these experiments, illustrating a consistent behavioral pattern of the proposed W2SC-Net model. The model achieves similar accuracy on the original test set and under low levels of distortion. But, as the magnitude of distortion increases, the performance gradually degrades. Specifically, the worst performance drops are 13.3% for resizing, 10.18% for JPEG compression, 13.94% for Gaussian noise, 26.53% for Gaussian blur, and 5.33% for histogram equalization.

To enhance robustness against anti-forensic operations, we extended the training of the proposed W2SC-Net model using additional data augmentation. Building upon the previously trained W2SC-Net (as explained in the Section IV-B), we continued training for an extra 30 epochs, incorporating anti-forensic operations as a data augmentation. The additional 30 epochs were practically sufficient to improve the robustness against anti-forensic attacks without overfitting, as the model was already fully trained on clean inpainted data. The additional augmentation process randomly applies one of the following transformations to each training image:

- Resizing: reduced to 90% of the original size
- JPEG Compression: quality factor of 85
- Gaussian Noise: standard deviation of 0.006

Table VI confirms the effectiveness of the additional training strategy in enhancing the model's resilience to anti-forensic attacks. The proposed enhanced version demonstrates remarkable robustness by maintaining consistently high detection accuracy (AUC above 94%) even under the strongest levels of distortion. This improvement is not limited to scenarios involving anti-forensic operations; it also extends to the original test set without any distortions, where the enhanced version slightly outperforms the baseline model by 0.54%.

V. CONCLUSION

In this paper, we propose W2SC-Net, a hybrid CNN–Swin model designed to detect and localize any type of inpainting that is applied to manipulate images. W2SC-Net achieves strong performance in image forensics by integrating an enhancement block, a dual-stream encoder, a hierarchical decoder with skip connections, and a joint BCE–Dice loss function. Experimental results demonstrate not only the superiority of W2SC-Net against state-of-the-art approaches but also its detection stability across a variety of inpainting methods. Moreover, the W2SC-Net and its introduced enhanced variant illustrate robustness to variations in manipulation ratio and resistance to anti-forensic operations. Collectively, these results validate the proposed model as a robust and effective solution for real-world inpainting forensic applications. For future work, the proposed model requires further improvement in its generalizability to adapt to the rapid progress of inpainting techniques. Developing a large-scale dataset that employs a diversity of inpainting methods in training and testing is critical to improve the detection accuracy. Despite the high detection accuracy achieved by the proposed enhanced version of W2SC-Net under anti-forensic attacks, it does not fully address this challenge when the operations become more sophisticated, such as when applying combined or advanced techniques. Finally, optimizing the architecture can help reduce the number of unnecessary parameters.

REFERENCES

[1] J.-B. Huang, S. B. Kang, N. Ahuja, and J. Kopf, "Image completion using planar structure guidance," *ACM Trans. Graph. (TOG)*, vol. 33, no. 4, pp. 1–10, 2014.

[2] J. Herling and W. Broll, "High-quality real-time video inpainting with PixMix," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 6, pp. 866–879, 2014.

[3] Q. Guo, S. Gao, X. Zhang, Y. Yin, and C. Zhang, "Patch-based image inpainting via two-stage low rank approximation," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 6, pp. 2023–2036, 2017.

[4] M. Bertalmio, A. Bertozzi, and G. Sapiro, "Navier-stokes, fluid dynamics, and image and video inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1. IEEE, 2001, pp. I–I.

[5] A. Telea, "An image inpainting technique based on the fast marching method," *J. Graph. Tools*, vol. 9, no. 1, pp. 23–34, 2004.

[6] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*. IEEE, 2018, pp. 5505–5514.

[7] Z. Yan, X. Li, M. Li, W. Zuo, and S. Shan, "Shift-net: Image inpainting via deep feature rearrangement," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Springer-Verlag, 2018, p. 3–19.

[8] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro, "Image inpainting for irregular holes using partial convolutions," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Springer International Publishing, 2018, pp. 89–105.

[9] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Free-form image inpainting with gated convolution," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*. IEEE, 2019, pp. 4471–4480.

[10] H. Wu, J. Zhou, and Y. Li, "Deep generative model for image inpainting with local binary pattern learning and spatial attention," *IEEE Trans. Multimedia*, vol. 24, pp. 4016–4027, 2022.

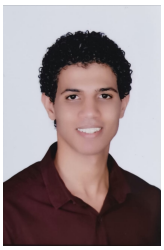
[11] T. Yu, Z. Guo, X. Jin, S. Wu, Z. Chen, W. Li, Z. Zhang, and S. Liu, "Region normalization for image inpainting," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 07, 2020, pp. 12 733–12 740.

[12] T. Ružić and A. Pižurica, "Context-aware patch-based image inpainting using markov random field modeling," *IEEE Trans. Image Process.*, vol. 24, no. 1, pp. 444–456, 2015.

[13] 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, no. 3, pp. 1172–1185, 2021.

[14] K. Liu, J. Li, and S. S. Hussain Bukhari, "Overview of image inpainting and forensic technology," *Secur. Commun. Netw.*, vol. 2022, no. 1, p. 9291971, 2022.

[15] Z. Liang, G. Yang, X. Ding, and L. Li, "An efficient forgery detection algorithm for object removal by exemplar-based image inpainting," *J. Vis. Commun. Image Represent.*, vol. 30, pp. 75–85, 2015.
[16] D. T. Trung, A. Beghdadi, and M.-C. Larabi, "Blind inpainting forgery detection," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*. IEEE, 2014, pp. 1019–1023.
[17] H. Li, W. Luo, and J. Huang, "Localization of diffusion-based inpainting in digital images," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 12, pp. 3050–3064, 2017.
[18] Y. Zhang, T. Liu, C. Cattani, Q. Cui, and S. Liu, "Diffusion-based image inpainting forensics via weighted least squares filtering enhancement," *Multimed. Tools Appl.*, vol. 80, pp. 30 725–30 739, 2021.
[19] X. Zhu, Y. Qian, X. Zhao, B. Sun, and Y. Sun, "A deep learning approach to patch-based image inpainting forensics," *Signal Process. Image Commun.*, vol. 67, pp. 90–99, 2018.
[20] M. Lu and S. Niu, "A detection approach using lstm-cnn for object removal caused by exemplar-based image inpainting," *Electronics*, vol. 9, no. 5, p. 858, 2020.
[21] N. Kumar and T. Meenpal, "Semantic segmentation-based image inpainting detection," in *Proc. Innov. Electr. Electron. Eng. (ICEEE)*. Springer, 2020, pp. 665–677.
[22] Y. Zhang, F. Ding, S. Kwong, and G. Zhu, "Feature pyramid network for diffusion-based image inpainting detection," *Inf. Sci.*, vol. 572, pp. 29–42, 2021.
[23] H. Li and J. Huang, "Localization of deep inpainting using high-pass fully convolutional network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2019, pp. 8301–8310.
[24] A. Li, Q. Ke, X. Ma, H. Weng, Z. Zong, F. Xue, and R. Zhang, "Noise Doesn't Lie: Towards universal detection of deep inpainting," in *Proc. 30th Int. Joint Conf. Artif. Intell. (IJCAI-21)*. International Joint Conferences on Artificial Intelligence Organization, 8 2021, pp. 786–792.
[25] C. Xiao, F. Li, D. Zhang, P. Huang, X. Ding *et al.*, "Image inpainting detection based on high-pass filter attention network." *Comput. Syst. Sci. Eng.*, vol. 43, no. 3, 2022.
[26] Z. Chen, Y. Zhang, Y. Wang, J. Tian, and F. Wu, "Robust image inpainting forensics by using an attention-based feature pyramid network," *Applied Sciences*, vol. 13, no. 16, p. 9196, 2023.
[27] H. Wang, X. Zhu, H. Sun, T. Qian, and Y. Chen, "A multi-path inpainting forensics network based on frequency attention and boundary guidance," *Electronics*, vol. 12, no. 14, p. 3192, 2023.
[28] S. A. E. Aly, O. Emara, H. Mahmoud, and S. Bekhet, "Detecting image forgery: a deep learning framework with feature pyramid integration for inpainting detection," *Neural Computing and Applications*, pp. 1–17, 2025.
[29] Y. Wu, W. AbdAlmageed, and P. Natarajan, "Mantra-net: Manipulation tracing network for detection and localization of image forgeries with anomalous features," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 9543–9552.
[30] X. Wang, S. Niu, and H. Wang, "Image inpainting detection based on multi-task deep learning network," *IETE Tech. Rev.*, vol. 38, no. 1, pp. 149–157, 2021.
[31] L. Hu, Y. Li, J. You, R. Liang, and X. Li, "An edge-aware transformer framework for image inpainting detection," in *Proc. Int. Conf. Artif. Intell. Secur.* Springer, 2022, pp. 648–660.
[32] W. Yang, R. Cai, and A. Kot, "Image inpainting detection via enriched attentive pattern with near original image augmentation," in *Proc. ACM Int. Conf. Multimedia (MM)*, 2022, pp. 2816–2824.
[33] H. Wang, X. Zhu, C. Ren, L. Zhang, and S. Ma, "A frequency attention-based dual-stream network for image inpainting forensics," *Mathematics*, vol. 11, no. 12, p. 2593, 2023.
[34] F. Xiao, Z. Zhang, and Y. Yao, "CTNet: hybrid architecture based on cnn and transformer for image inpainting detection," *Multimed. Syst.*, vol. 29, no. 6, pp. 3819–3832, 2023.
[35] X. Ding, Y. Deng, Y. Zhao, and W. Zhu, "AFTLNet: An efficient adaptive forgery traces learning network for deep image inpainting localization," *Journal of Information Security and Applications*, vol. 84, p. 103825, 2024.
[36] M. Liu, X. Di, and M. Liao, "Image inpainting detection via dual guidance of uncertainty and precise boundary information," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 35, no. 10, pp. 10 305–10 315, 2025.
[37] Y. Yao, T. Han, S. Jia, and S. Lyu, "Dense feature interaction network for image inpainting localization," *IEEE Trans. Inf. Forensics Secur.*, vol. 20, pp. 1636–1648, 2025.
[38] S. Liu, J. Chen, X. Ding, and G. Yang, "Progressive reverse attention network for image inpainting detection and localization," *Computer Vision and Image Understanding*, p. 104407, 2025.
[39] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2021, pp. 10 012–10 022.
[40] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1452–1464, 2018.
[41] T. Gloe and R. Böhme, "The 'dresden image database' for benchmarking digital image forensics," in *Proc. ACM Symp. Appl. Comput.*, 2010, pp. 1584–1590.
[42] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," in *Int. Conf. Learn. Represent. (ICLR)*, 2018.
[43] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*. Ieee, 2009, pp. 248–255.
[44] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. [Online]. Available: https://doi.org/10.48550/arXiv.1412.6980
[45] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020. [Online]. Available: https://doi.org/10.48550/arXiv.2010.11929
[46] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 1290–1299.
[47] B. Bayar and M. C. Stamm, "Constrained convolutional neural networks: A new approach towards general purpose image manipulation detection," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 11, pp. 2691–2706, 2018.
[48] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 2980–2988.
[49] N. Abraham and N. M. Khan, "A novel focal tversky loss function with improved attention u-net for lesion segmentation," in *Proc. IEEE Int. Symp. Biomed. Imaging (ISBI)*. IEEE, 2019, pp. 683–687.



Mohamed Fathy Mohamed received his B.Sc. degree in Computer Science from the Faculty of Computers and Information, Luxor University, Egypt. He is currently a Teaching Assistant at the same faculty. His academic journey includes contributions to several AI-related projects and participation in problem-solving programming contests. He is also pursuing an M.Sc. degree in Computer Science at the Faculty of Computers and Artificial Intelligence, Helwan University, Egypt. His research interests include artificial intelligence, machine learning, deep learning, computer vision, image processing, and computational optimization.

Hala Abdel-Galil ElSayed is a Professor of Artificial Intelligence at the Computer Science Department, Faculty of Computers and Artificial Intelligence, Helwan University, Cairo, Egypt. She served as Head of the Computer Science Department from 2015 to 2024. She received her B.Sc. and M.Sc. degrees in Computer Science from Ain Shams University and her Ph.D. in Scientific Computations from Cairo University. Her research interests include artificial intelligence, neural networks, computational modeling, and decision support systems.

Soha Ahmed Ehssan Aly received her B.Sc. and M.Sc. degrees in Computer Science from the Faculty of Computers and Artificial Intelligence, Helwan University, Cairo, Egypt. She earned her Ph.D. from the School of Computer Science and Informatics, Cardiff University, Wales, UK, where she was a member of the Informatics Research Group. She is currently an Assistant Professor at the Department of Computer Science, Helwan University. Her research interests include data science, natural language processing, machine learning, geo-social analysis, and image processing.