Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.061252

ARTICLE

Robust Image Forgery Localization Using Hybrid CNN-Transformer Synergy Based Framework

Sachin Sharma[1,2,*], **Brajesh Kumar Singh**[3] **and Hitendra Garg**[2]

[1]Department of Computer Science, Abdul Kalam Technical University, Lucknow, 226031, India
[2]Department of Computer Engineering & Applications, GLA University, Mathura, 281406, India
[3]Department of Computer Science, Raja Balwant Singh Engineering Technical Campus, Agra, 283105, India

*Corresponding Author: Sachin Sharma. Email: sachin.sharma@gla.ac.in

Received: 20 November 2024; Accepted: 21 January 2025; Published: 06 March 2025

ABSTRACT: Image tampering detection and localization have emerged as a critical domain in combating the pervasive issue of image manipulation due to the advancement of the large-scale availability of sophisticated image editing tools. The manual forgery localization is often reliant on forensic expertise. In recent times, machine learning (ML) and deep learning (DL) have shown promising results in automating image forgery localization. However, the ML-based method relies on hand-crafted features. Conversely, the DL method automatically extracts shallow spatial features to enhance the accuracy. However, DL-based methods lack the global co-relation of the features due to this performance degradation noticed in several applications. In the proposed study, we designed FLTNet (forgery localization transformer network) with a CNN (convolution neural network) encoder and transformer-based attention. The encoder extracts local high-dimensional features, and the transformer provides the global co-relation of the features. In the decoder, we have exclusively utilized a CNN to upsample the features that generate tampered mask images. Moreover, we evaluated visual and quantitative performance on three standard datasets and comparison with six state-of-the-art methods. The IoU values of the proposed method on CASIA V1, CASIA V2, and CoMoFoD datasets are 0.77, 0.82, and 0.84, respectively. In addition, the F1-scores of these three datasets are 0.80, 0.84, and 0.86, respectively. Furthermore, the visual results of the proposed method are clean and contain rich information, which can be used for real-time forgery detection. The code used in the study can be accessed through URL: https://github.com/ajit2k5/Forgery-Localization (accessed on 21 January 2025).

KEYWORDS: Image; tampering; convolution neural network (CNN); hybrid; transformer; localization

1 Introduction

The advancement of digital platforms has increased image manipulation. Today, we share, create and download image from the internet. The intruder easily tampered the shared images, which posed a substantial challenge in various industries [1]. In addition, due to the large-scale availability of advanced image editing tools, fake images can be created. These fake images are challenging to discriminate as they look natural, potentially causing severe consequences [2]. Due to this, a lack of trust can be noticed in the media organization, and the credibility of visual journalism is at risk. Moreover, tampered images pose a significant challenge for legal settings as they can be used for evidence in several crimes [3].

In the manual process of image authenticity validation, the forensic team collects the images of evidence and examines the image characteristics to identify the anomalies that might indicate tampering [4].

One classical method is image pixel analysis, which involves an examination of the pixel's value to identify inconsistency in the image. The pixel value of the tampered region has different characteristics compared to the authentic region [5]. Another method is copy-move forgery detection, in which the expert aims to identify the replicated pixels or regions within the original image [6]. Experts detect differences in compression levels throughout the image in the ELA (Error Level Analysis) approach. Forged regions that have been compressed with a different compression level will stand out [7].

The manual methods have done a great job in image authenticity analysis. However, traditional image authenticity methods require highly expert people, and at the same time, it is time-consuming. In addition, these techniques largely depend on the expertise of the forensic analyst [8]. In addition, conventional techniques could be more efficient when dealing with low image resolution and images with excessive noise. Moreover, professional forgers can produce fake images that closely mimic authentic ones, posing challenges in their identification through traditional methods [9]. Several machine learning (ML) and deep learning (DL) techniques have been designed to address these constraints. The ML-based methods leverage manually designed texture, shape, and colour information to train algorithms. Consequently, the effectiveness of these approaches relies on the ability of the forensic professional [10]. Furthermore, deep learning-based systems automatically extract the image features to identify the tampered region [11]. Nevertheless, DL methods necessitate substantial data and a comprehensive correlation of global information [12]. This affects the method's effectiveness when dealing with low image resolution and images containing noise [13].

We designed a CNN-based encoder that focused on local spatial features. Furthermore, a transformer block is utilized in the encoder, which provides attention. In addition, the transformer block enhanced the global co-relation of the spatial features. Further, we have used purely CNN block in the decoder to upsample the spatial features. In addition, we utilized cascading upsampling blocks, with each deconvolution block consisting of a 2x upsampling operator to generate the tampered mask. We evaluated the model's performance on the CASIA V1, CASIA V2 and CoMoFoD datasets. The FLTNet achieved much better quantitative and visual results than several other methods.

The significant contribution of the method is as follows:

(a) We extracted spatial features using CNN-based encoders and provided attention using transformer block to generate fine-grained localized images.
(b) The MSA in classical ViT cannot capture fine grained local spatial features. Therefore, we replaced it with W-MSA in the transformer block to extract enhanced local attention within a window, followed by the SW-MSA.
(c) The proposed FLTNet is experimentally evaluated on the three standard datasets and compared with six state-of-the-art methods. The visual and quantitative results are superior compared to other methods.

The rest of the manuscript is organized as follows:

The detailed literature reviews are added in Section 2, whereas, in Section 3, the FLTNet's architecture and loss functions are explained. Furthermore, in Section 4, we have added an analysis of the quantitative and visual results of the proposed method and several other methods. and finally, the conclusion and future scope are provided in Section 5.

2 Literature Review

The image forgery localization is a vital problem for digital image security. Several methods have been utilized to identify and locate the forged region in the image. In this regard, Cozzolino et al. [14] proposed a localization framework that combines sensor noise, patch-matching, and machine learning tools, fusing their binary masks using reliability indexes. Initial experiments on the training set show high

localization accuracy and helpful visual clues. Yarlagadda et al. [15] proposed an algorithm for detecting and localizing satellite image forgery, focusing on scenarios where objects are added or removed. The algorithm uses a generative adversarial network (GAN) to learn features from pristine satellite images. A one-class support vector machine (SVM) is then trained on these features to identify anomalies and detect image forgeries. Kumar et al. [16] utilized a deep semantic image-inpainting algorithm to generate a synthetic forged dataset. In addition, an unsupervised domain adaptation network was employed to detect image copy-move forgery localization.

Le et al. [17] explore image forgery localization, focusing on demosaicing artefacts. They improve authentication and localization by linking colour filter array patterns, demosaicing algorithm estimation, and statistical analysis of artefacts. Bi et al. [18] proposed a fake-to-realistic transform generator for tampered region localization. Through adversarial training, colouring transforms their method and automatically suppresses tampering artefacts in forged images. Cozzolino et al. [19] utilized a CNN-based method. They trained it to forge an image dataset distinguishing patches from the same camera model. Bunk et al. [20] proposed an image tampering detection method using resampling features and deep learning. Guillaro et al. [21] introduced TruFor, a flexible forensic framework designed to identify several types of image manipulation. Guo et al. [22] suggested a hierarchical fine-grained method for learning IFDL representation. They improve the algorithm's potential for understanding various levels of forgery traits in modified images. Hao et al. [23] introduced TransForensics, a novel technique for identifying image forgeries, which draws inspiration from Transformers. This framework comprises compact self-attention encoders that capture overall context and pairwise interactions among nearby patches.

Bianchi et al. [24] developed an algorithm to identify double JPEG compression in manipulated images without manually selecting certain regions. The system utilizes an enhanced statistical model to automatically calculate a likelihood map for every 8×8 block, which indicates the possibility of double compression. Ferrara et al. [25] introduced a forensic tool to distinguish between authentic and manipulated areas in digital camera images by utilizing demosaicking artefact assessment at a local level and employing a statistical model, which can determine the likelihood of tampering for each 2×2 image block. Li et al. [26] presented a method that improves the accuracy of identifying forged areas by combining tampering possibility maps. Singh et al. [27] implemented a CNN to identify fake images on social media networks. The initial layer of the model employs high pass filters to initialize weights, improving convergence speed and accuracy. Dua et al. [28] proposed an advanced methodology for examining tampering in JPEG compressed images, covering splicing and copy-move forgery. The technique utilizes block processing to verify and determine the exact locations of modified areas. Rao et al. [29] developed a novel approach for detecting and identifying image forgery. They combined CNN with a multi-semantic Conditional Random Field (CRF) attention model by utilizing boundary transition artefacts.

Jabeen et al. [30] proposed a multimodal system that utilizes CNN for forgery detection. InceptionV3 is used to extract spatial features from manipulated regions. Liu et al. [31] proposed a Fusion-Net for locating tampered areas by tracing borders. They trained on specific splicing forgery types using Base-Net; Fusion-Net discerns whether an image block is synthesized from different origins after fine-tuning with a minimal number of images. Lin et al. [32] focused on forgery localization using photo-response non-uniformity (PRNU) noise. They proposed a segmentation model that leverages local homogeneity to address the challenges of existing approaches. Rao et al. [33] proposed a self-supervised domain adaptation network for JPEG-resistant forgery localization. Comprising a vanilla architecture backbone and a Compression Approximation Network (ComNet), it mimics JPEG compression through self-supervised learning. Kumar et al. [34] applied CNN model inspired by the VGG16 for the detection of the multiple forgery in the video. Their model achieved 91% accuracy on the VIFFD dataset.

3 Proposed Method

In the proposed study, we localized the label mask of the tampered region. Unlike Unet [35], in which encoder and decoder are CNN blocks, which downsample and upasmple the image to construct the mask. We provided attention to the features extracted in the encoder block through transformer. The overall architecture of the proposed method is shown in Fig. 1.

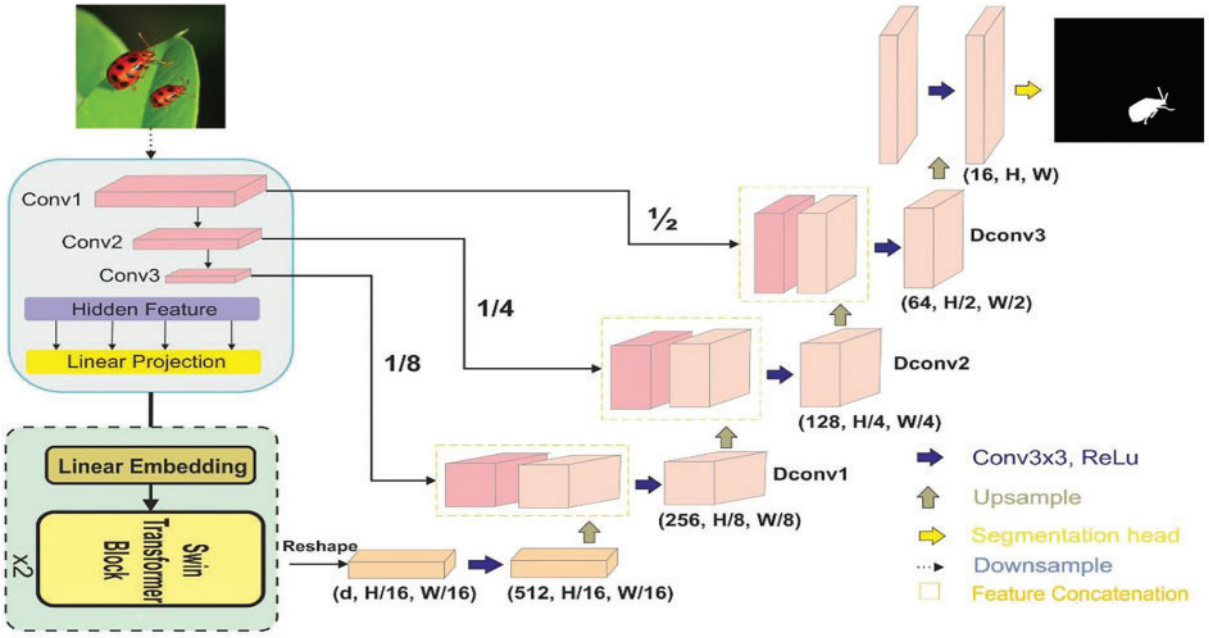

Figure 1: The architecture of the FLTNet for tempered image localization

3.1 Convolution Encoder with Transformer Attention

The input image $I \in \mathbb{R}^{H\times W\times 3}$is passed to the convolution block (Conv1) of convolution size 64 having 3 × 3 filter, followed by stride of size 2 and ReLu activation. The ReLu activation squashes the negative value and produces output between zero and positive values as follows:

$$F(y) = \left\{ \begin{array}{l} 0, y < 0 \\ y, y >= 0 \end{array} \right\}. \tag{1}$$

Subsequently, a max-pooling of window size 2 × 2 is applied to downsample the spatial features. The next Conv2 block has convolution size 128 with filter size 3 × 3 followed by stride of size 2 and ReLu activation. For the Conv3 block, we have utilized convolution size 256, followed by the stride of size 2, ReLu activation and max-pooling of 2 × 2. The detailed architecture of the convolution blocks used in the encoder is shown in Fig. 2.

The spatial features extracted from the convolutions blocks are linearly projected, and a batch embedding is performed on a 1 × 1 features map obtained from CNN blocks to generate the tokens as follows:

$$Z_0 = \left[y_p^1 E; y_p^1 E y_p^2; y_p^3 E; \cdots\cdots; y_p^M E \right] + E_{pos}, \tag{2}$$

where y_p = Vectorized patches of D-dimensional, $E \in \mathbb{R}^{(p^2 \cdot C) \times D}$ = Patch embedding projection and $E_{pos} \in \mathbb{R}^{M \times D}$ = Position encoding.

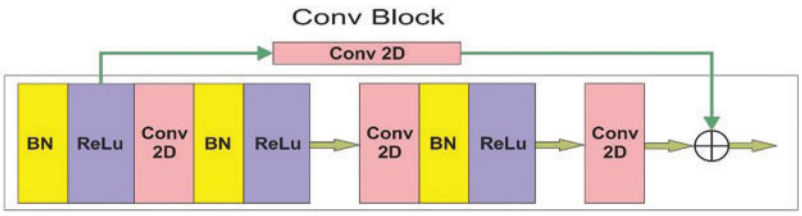

Figure 2: The architecture of the convolution blocks used in the encoder

The classical transformer encoder consists of MSA (multihead self-attention), MLP (multi-layer perceptron) and LN (layer norm). The output of the *J*th layer is calculated as follows:

$$\begin{aligned} \tilde{Z}^J &= \mathrm{MSA}\left(\mathrm{LN}\left(Z^{J-1}\right)\right) + Z^{J-1}, \\ Z^J &= \mathrm{MLP}\left(\mathrm{LN}\left(\tilde{Z}^J\right)\right) + \tilde{Z}^J. \end{aligned} \tag{3}$$

In MSA, the relationship of each token is calculated with other tokens that increase the complexity to quadratic time. Hence, this makes it less acceptable for high-resolution image segmentation and classification tasks [36]. We utilized a W-MSA (window multi-head self-attention) and SW-MSA (shifted-window multi-head self-attention) to reduce the computation burden. The architecture of the classical and swing transformer block is shown in Fig. 3

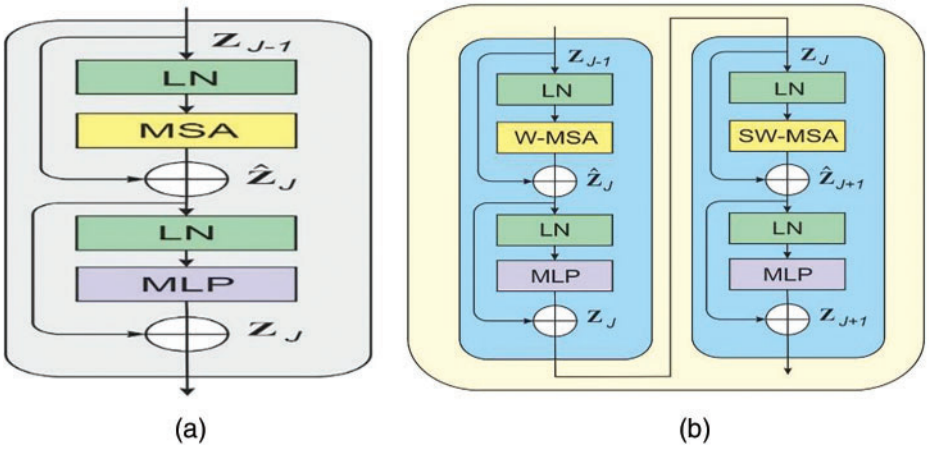

Figure 3: The classical and Swin blocks' architecture

In W-MSA, attention is calculated on the local widows of patch size S × S on the input features. The output of the *J*th layer is calculated as follows:

$$\begin{aligned} \tilde{Z}^J &= \mathrm{WMSA}\left(\mathrm{LN}\left(Z^{J-1}\right)\right) + Z^{J-1}, \\ Z^J &= \mathrm{MLP}\left(\mathrm{LN}\left(\tilde{Z}^J\right)\right) + \tilde{Z}^J. \end{aligned} \tag{4}$$

However, it suffers from a lack of interaction among local windows. To overcome the issue, SW-MSA is incorporated after the W-MSA module, as shown in Fig. 2b. The architecture of SW-MSA differs from the previous W-MSA layer, as it utilizes an effective batch processing strategy by cyclically shifting towards the upper left direction. Following this shift, a batch window in the feature map may consist of many non-adjacent sub-windows while maintaining the same number of batch windows. During the computation of self-attention in both W-MSA and SW-MSA, the similarity calculation considers the relative location bias inside local windows. After shifting, the output of the SW-MSA and MLP is calculated as follows:

$$\begin{aligned} \tilde{Z}^{J+1} &= \text{SWMSA}\left(\text{LN}\left(Z^{J}\right)\right) + Z^{J}, \\ Z^{J+1} &= \text{MLP}\left(\text{LN}\left(\tilde{Z}^{J+1}\right)\right) + \tilde{Z}^{J+1}. \end{aligned} \tag{5}$$

The SA of the W-MSA and SW-MSA blocks is defined as follows:

$$\begin{aligned} &Q = Z^{J}W_{Q}, K = Z^{J}W_{K}, V = Z^{J}W_{V}, \\ &\text{Attn}\left(Z^{J}\right) = \text{Softmax}\left(QK^{T}/\sqrt{d} + B\right)V \end{aligned} \tag{6}$$

where Q, K, and V are the query, key and value of the matric WQ, WK, and WV, respectively, and B is the relative bias position of the transformer layer. The hidden features sequence obtained from the transformer block is reshaped and passed to the decoder block.

3.2 The Convolution-Based Decoder

In the decoder block, we have utilized only CNN blocks. Each CNN block consists of 3 × 3 filters followed by BN and ReLu activation. The sizes of Dconv1, Dconv2, and Deconv3 are 256, 128 and 64, respectively. The architecture of the deconvolution blocks used in the decoder blocks is shown in Fig. 4.

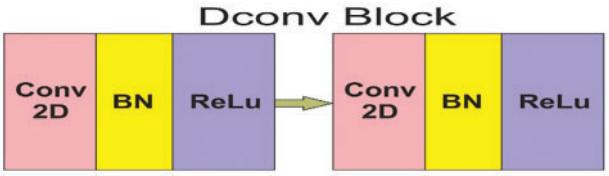

Figure 4: The architecture of the deconvolution blocks used in the decoder

The features obtained from the Conv1, Conv2 and Conv3 are concatenated with the Dconv1, Dconv2 and Deconv3, respectively. Further, to obtained the image of size HxW from the $\frac{H}{p} \times \frac{W}{p}$ patches. We applied a cascading upsampling block, with each block consisting of a 2x upsampling operator. Finally, the 1 × 1 convolution and sigmoid activation are applied to obtain the tampered region of the image. The proposed FLTNet has three major components: a CNN encoder to extract local spatial features, a ViT block for local and global attention and a decoder for upsampling the features. Several studies suggest that CNN may miss the boundary and edge region. To complement the issue, we applied a ViT block based on the SW-MSA, which provides local and global contextual information through shift-window interaction. Furthermore, a skip connection from the encoder is applied to the decoder block that passes information. The decoder block upsamples the feature map, generating a detailed localized forged image. Similar to our work, the recent trends of a hybrid CNN-transformer framework utilized pre-trained ResNet50 and a ViT encoder having MSA (multi-head self-attention), which computes attention pairwise in the token and provides global attention. Due to this, it misses the local contextual features, and computation cost rises to quadratic time complexity.

3.3 The Loss Function

The loss for all experiments was calculated using a composite of the dice and binary focal loss functions. The similarity between the ground truth (GT) and the segmented picture was determined using the dice loss. The binary focal loss penalizes hard-to-classify regions more strongly than easy-to-classify regions. The mathematical description of the loss function is as follows:

$$\mathrm{DSC} = \frac{(2|\mathrm{M} \cap \mathrm{N}|)}{(|\mathrm{M}| + |\mathrm{N}|)}. \tag{7}$$

Here, M = Labelled GT, N = Segmented image. The BFL (binary focal loss) is defined as follows:

$$\mathrm{BFL}(\mathrm{X}, \mathrm{P}) = -\beta \mathrm{X} (1 - \mathrm{P})^{\alpha} \log(\mathrm{P}) - (1 - \mathrm{y}) \mathrm{P}^{\alpha} \log(1 - \mathrm{P}) \tag{8}$$

Here, α = Focusing parameters, β = Trade-off between recall and precision, X = Binary class labelled and P = Estimated probability of tampered class.

$$\mathrm{Loss} = \mathrm{DSC} + (1 \times \mathrm{BLF}) \tag{9}$$

Here, DSC is the dice coefficient for foreground and background pixels.

4 Results

In this section, we have discussed quantitative and visual results on three datasets, CASIA V1, CASIA V2 and CoMoFoD, of the proposed model and six other models.

4.1 Dataset

In this section, we have discussed the three datasets CASIA V1, CASIA V2 and CoMoFoD.

4.1.1 CASIA V1 Dataset

The CASIA V1 dataset has 1721 images with a resolution of 384 × 256 pixels. It has 921 tampered and 800 authentic images stored in JPEG format. Most of the images in the authentic class are taken from the Corel image dataset. Meanwhile, tampered images are generated from authentic images through copy-paste and crop operations [37]. The GT (Ground Truth) images are generated using the procedure described in the literature [38]. A sample image and its ground truth are shown in Fig. 5.

Figure 5: The original image, forged image and its GT. (a) Original image (b) Forged image and (c) GT

4.1.2 CASIA V2 Dataset

The CASIA V2 dataset contains 5123 tampered and 7200 authentic images widely used in image forgery detection. It contains 9 categories of images with varying size ranges from 320 × 240 to 800 × 600 pixels. In

addition, CASIA V2 dataset images are stored in TIFF and BMP formats. The images in the authentic class are taken from the Corel image dataset. The GT (Ground Truth) images are generated using the procedure described in the literature [38]. The sample images are shown in Fig. 6.

Figure 6: The original image, forged image and its GT. (a) Original image (b) Forged image and (c) GT

4.1.3 The CoMoFoD Dataset

The CoMoFoD dataset consists of 200 images with a resolution of 512 × 512 pixels. Images are generated using five types of manipulations. Each category contains 40 images. After applying preprocessing, the dataset is enlarged to 8500 images. In the dataset, GT masks exhibit manipulated and source regions [39]. For our localization problem, changes were made in the image to generate GT masks of the manipulated region according to the method used for the CASIA V2 dataset. The sample images of the CoMoFoD dataset are shown in Fig. 7.

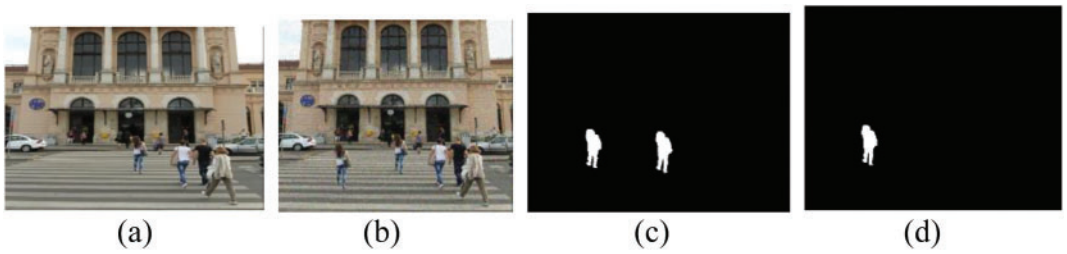

Figure 7: The original image, forged image and its masked and GT images. (a) Original image (b) Forged image (c) Masked image and (d) GT

4.2 Experimental Setup

We evaluated the proposed method on NVIDIA Quadro RTX-4000 GPU, which has 128 GB RAM and a dual graphics card of 8 GB. The script is written using Python 3.9. Each experiment is carried out for 200 epochs in a batch size of 64 using the sigmoid optimizer.

4.3 Quantitative Results

In the proposed study, we evaluated the performance of the FLTNet on the three datasets. To evaluate the performance on the CASIA V1 dataset for each experiment, images are reshaped to 300 × 300 pixels. After that, each model is trained for 200 epochs in a batch size of 64 by dividing the dataset randomly into 80% and 20% for training and validation. We compared EDTNet performance with SegNet [40], Unet [41], ManTra-Net [42], LSTM-EnDec [43], RGB-N [44], SPAN [45]. The SegNet is a based encoder and decoder network inspired by the VGG16. In SegNet, the encoder uses the first 13 layers of the VGG16, and the decoder consists of 4 convolution blocks. Each convolution block in the encoder is followed by a BN (batch normalization) stride and a max-poling layer. The Unit is also an encoder and decoder-based model designed

using 2D convolution blocks consisting of 23 layers. ManTra-Net is a self-supervised end-to-end network used for image forgery detection and localization. LSTM-EnDec is a hybrid model designed using LSTM (long short-term memory) and CNN-based encoder and decoder. The RGB-N utilized a ResNet-101 network as a backbone to extract spatial features. In addition, a pre-trained Faster-RCNN is used to enhance the image forgery localization. SPAN uses the VGG network as the backbone and 2D CNN for the spatial feature extraction. In addition, the mask is generated using a 2D convolution layer and sigmoid activation function.

Furthermore, the performance indicators IoU (Intersection over union), precision, recall, F1-score and loss are calculated as shown in Table 1. Table 1 shows that the RGB-N achieved the least IoU of 0.43 and loss of 0.76. Slightly improved precision and IOU values can be observed in SegNet. The second highest IoU and the precision value obtained by the SPAN. Meanwhile, the FLTNet achieved the IoU with 0.77 precision.

Table 1: Comparative performance on CASIA V1 dataset

Methods	Precision	Recall	F1-score	IoU	Loss
SegNet	0.57	0.49	0.53	0.47	0.72
Unet	0.52	0.46	0.49	0.44	0.85
ManTra-Net	0.69	0.73	0.71	0.65	0.61
LSTM-EnDec	0.72	0.69	0.70	0.67	0.57
RGB-N	0.45	0.49	0.47	0.43	0.76
SPAN	0.75	0.79	0.77	0.73	0.51
Proposed	0.78	0.83	0.80	0.77	0.37

Further, FLTNet performance on CASIA V2 dataset images is evaluated by reshaping them to 300 × 300 pixels. After that, each model is trained for 200 epochs in a batch size of 64 by dividing the dataset randomly into 80% and 20% for training and validation using a sigmoid optimizer. Furthermore, the performance indicators IoU, precision, recall, F1-score and loss are calculated as shown in Table 2. In Table 2, we can see that the precision of the RGB-N is 0.58, which is the lowest in the table. Unet has improved slightly to 0.63. The SegNet and ManTra-Net have almost identical precision values. The second-highest precision value was obtained using SPAN. Meanwhile, the proposed FLTNet achieved the highest precision. The recall value of the FLTNet is slightly low compared to SPAN. However, the loss is much lower than that of SPAN.

The FLTNet performance was evaluated on the CoMoFoD dataset under the same experimental conditions that have been used for the CASIA V2 dataset. The performance indicators IoU, precision, recall, F1-score and loss on the CoMoFoD dataset are shown in Table 3. We can see that the F1-score of the Unet is 0.58 lowest in the table. Meanwhile, RGB-N achieved an F1-score of 0.61. The loss of the Unet and RGB-N are very close. The SPAN and FLTNet have an F1-score of 0.84 and 0.86, respectively. At the same time, the loss of these two models is 0.27 and 0.24, respectively.

Table 2: Comparative performance on CASIA V2 dataset

Methods	Precision	Recall	F1-score	IoU	Loss
SegNet	0.72	0.65	0.68	0.66	0.63
Unet	0.63	0.69	066	0.61	0.56
ManTra-Net	0.71	0.78	0.74	0.79	0.49
LSTM-EnDec	0.75	0.81	0.78	0.71	0.43

(Continued)

Table 2 (continued)

Methods	Precision	Recall	F1-score	IoU	Loss
RGB-N	0.58	0.62	0.60	0.56	0.61
SPAN	0.79	0.84	0.81	0.77	0.57
Proposed	0.83	0.85	0.84	0.82	0.32

Table 3: Comparative performance on CoMoFoD dataset

Methods	Precision	Recall	F1-score	IoU	Loss
SegNet	0.67	0.71	0.69	0.64	0.52
Unet	0.57	0.60	0.58	0.53	0.53
ManTra-Net	0.76	0.73	0.74	0.71	0.41
LSTM-EnDec	0.71	0.76	0.78	0.75	0.31
RGB-N	0.57	0.67	0.61	0.52	0.56
SPAN	0.83	0.86	0.84	0.71	0.27
Proposed	0.78	0.85	0.86	0.84	0.24

The CASIA V2 dataset contains images of varying resolution, and the tampered region is relatively large compared to CoMoFoD. The CoMoFoD dataset contains images of 512 × 512 resolution and 5 categories of tampering. We presented the quantitative performance of the FLTNet on the CASIA V2 and CASIA V3 in Tables 2 and 3, respectively. The performance of the model is relatively consistently high on both datasets. The FLTNet contains a multiscale CNN encoder for high-resolution spatial and coarse-grain local features. In addition, the ViT encoder provides local and global attention through W-MSA and SW-MSA to capture detailed information from the complex tampered region. Moreover, the hybrid loss function uses dice loss to preserve spatial coherence and binary focal loss to punish hard-to-classify regions. Overall, the multiscale CNN-based encoder and ViT with local and global attention focused on the edge and boundary region and CNN decoder upsample the feature, which leads to consistent performance across diverse datasets.

4.4 Visual Results

The visual results on CASIA V1, CASIA V2 and CoMoFoD are shown in Figs. 8–10. In Fig. 8, we can see that the masked image of the SegNet has a high level of noise. Since the SegNet encoder and decode are based on CNN, which ignores the boundary region features. The Unet generates a slightly improved localized image; mantra-net adopted a self-supervised technique to enhance the visual quality. However, the noise level is high. Furthermore, LSTM-EnDec utilized LSTM and CNN-based encoders and decoders that enhance the spatial features and co-relation of the features. The RGB-N utilized ResNet-101 as the backbone and a single Faster-RCNN to reduce noise. The SPAN utilized a pyramid structure and VGG16 to minimize the noise and generate a visual map better than the previous one. In the proposed FLTNet, we used a CNN-based encoder and transformer-based attention. The transformer attention block is based on sliding to provide MHA that improves the edge and corner features. The localized masked image has less noise and is closer to GT. In the decoder, we utilized multiple convolution layers for local spatial feature enhancement and cascade upsampling for reconstruction of the masked image. We applied a 2x upsampling operator to preserve the edge and boundary region. In addition, a skip connection from the encoder to the decoder block preserves the forged image's high-resolution texture. The Unet and SegNet contain CNN blocks as encoders and decoders. Due to this, noises can be transmitted through a skip connection. In addition, the

decoder block in the SegNet relies on the max-pooling operation, which misses the detailed information about the objects.

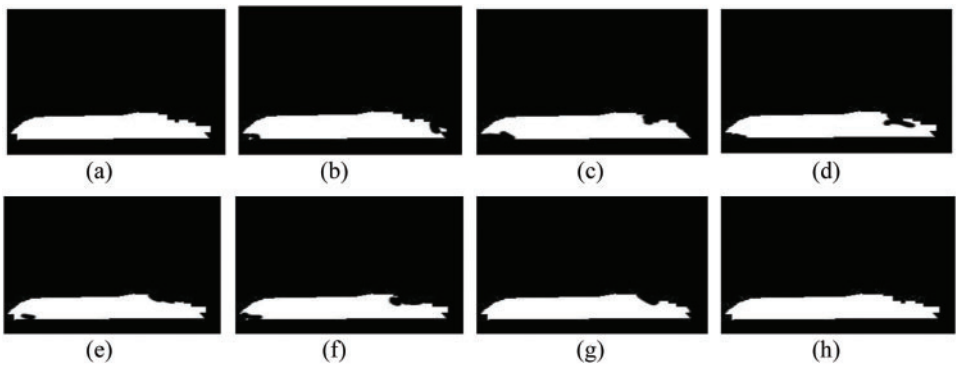

Figure 8: The visual map on CASIAV1 dataset (a) GT (b) SegNet (c) Unet (d) ManTra-Net (e) LSTM-EnDec (f) RGB-N (g) SPAN and (h) FLTNet

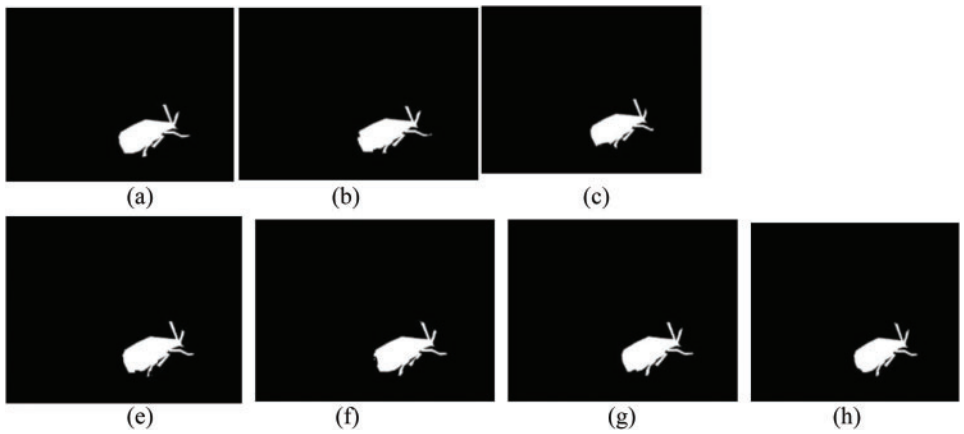

Figure 9: The visual map on CASIAV2 dataset (a) GT (b) SegNet (c) Unet (d) ManTra-Net (e) LSTM-EnDec (f) RGB-N (g) SPAN and (h) FLTNet

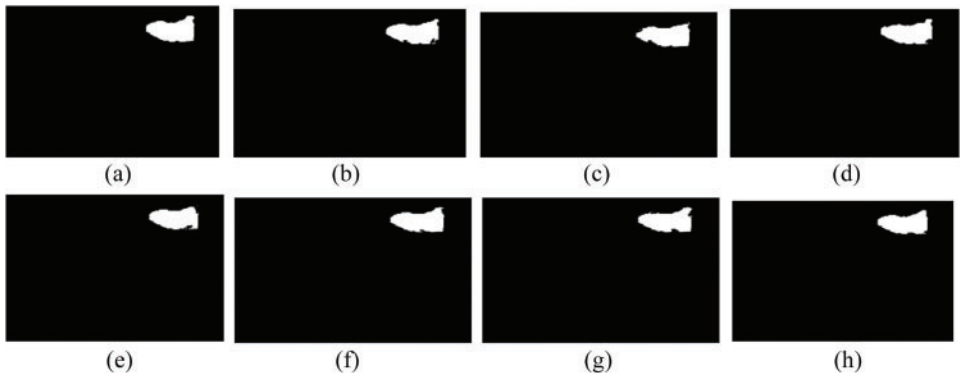

Figure 10: The visual map on CoMoFoD dataset (a) GT (b) SegNet (c) Unet (d) ManTra-Net (e) LSTM-EnDec (f) RGB-N (g) SPAN and (h) FLTNet

We can notice that the visual results of the FLTNet are close to GT and contain better edge and boundary regions compared to SPAN and RGB-N. The RGB-N utilized ResNet-101 as the backbone and a single Faster-RCNN to reduce noise. However, the RGB-N visual results contain noises, where the texture difference between forged and authentic images is less. The SPAN utilized a pyramid structure and VGG16 to minimize the noise. However, it struggles in the small tampered region due to self-attention. In the FLTNet, we utilized a CNN encoder. We provided attention using W-MSA and SW-MSA, which provided local and global attention to focus on the boundary and edge region of the tampered images. The detailed information and sharp boundaries in the visual results can provide better insight into the tempered images to the expert.

4.5 The Training Loss

We plotted the training loss of the FLTNet on CASIA V1, CASIA V2, and CoMoFoD datasets for 200 epochs, as shown in Fig. 11. In Fig. 11a, we can observe that model loss is initially very high. It started gradually decreasing after 25 epochs. In addition, several high and low peaks can be observed between 25 to 165 epochs. However, these peaks have a value below 0.50. After 175 epochs, it reaches close to 0.1. In Fig. 11b, FLTNet loss is initially more than 0.5. After 25 epochs, it reaches below 0.4. After that, it gradually decreased and reached close to 0.2 after 75 epochs. The training loss of the model on the CoMoFod dataset shown in Fig. 11c initially has several high and low peaks. After 50 epochs, it started decreasing and reached below 0.25. Finally, it reaches close to zero after 190 epochs.

4.6 Ablation Study

We rigorously anlayzed the effect of the different components of the FLTNet on three datasets. When only CNN CNN-based encoder and decoder are applied to three datasets, the IoU value is relatively less. Furthermore, the inclusion of the CNN+ViT in the encoder and pure CNN as the decoder improves the IoU value by almost 1%. In addition, we replaced MSA with W-MSA in the ViT, and the IoU value increases in the three datasets. In a final experiment, we can see in Table 4, that the highest value of IoU is obtained on the CASIA V1, CASIA V2 and CoMoFoD datasets. The MSA (multi-head self-attention) compute attention pairwise in the token and provides global attention. At the same time, W-MSA calculates local spatial attention using a no-overlapping window; due to this, it misses the global context. Therefore, we applied SW-MSA along with W-MSA, which computes attention by shifting the window cyclically and ensures spatial features from neighbor windows are brought in the same local window. In this way, both local and global feature interactions are enhanced in the ViT block.

We performed an ablation study on the IMD2020 Real-Life Manipulated Images dataset. This dataset contains 2000 real-world diverse manipulated images. For each image, their forged image and corresponding mask are provided for localization in the manipulated region [46]. We utilized the same experimental condition discussed in Section 4.2, and the quantitative results presented in Table 5. Table shows that the model achieved IoU and F1-score of 0.74 and 0.78, respectively.

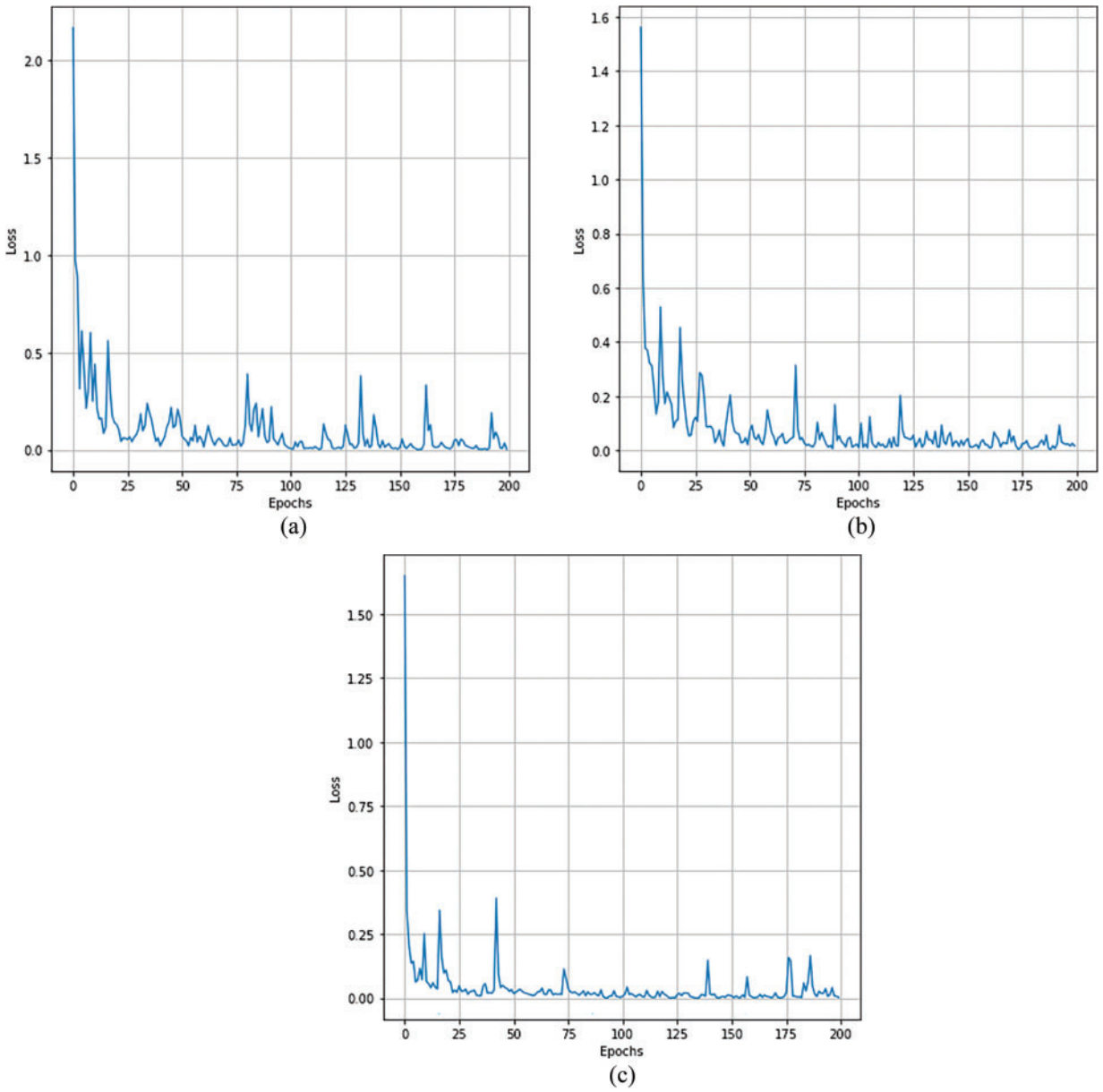

Figure 11: The training loss on CASIA V1, CASIA V2 and CoMoFoD dataset

Table 4: Effect of different component on performance

Components	CASIA V1	CASIA V2	CoMoFoD
	IoU	IoU	IoU
CNN (Enoder) +CNN (decoder)	0.73	0.78	0.81
CNN+ViT+MSA (Encoder)+CNN (decoder)	0.74	0.79	0.82
CNN+ViT+W-MSA (Encoder)+CNN (decoder)	0.75	0.81	0.83
CNN+ViT+W-MSA+SW-MSA (Encoder)+CNN (decoder)	0.77	0.82	0.84

Table 5: Performance of the proposed model on IMD2020 dataset

Precision	Recall	F1-score	IoU	Loss
0.76	0.80	0.78	0.74	0.51

Furthermore, we depict the visual results of the proposed model in Fig. 12. We can observe that the output of the FLTNet is close to GT. However, some portions of the top right edges need to be sharpened. This can be done by utilizing an attention mechanism to decoder block.

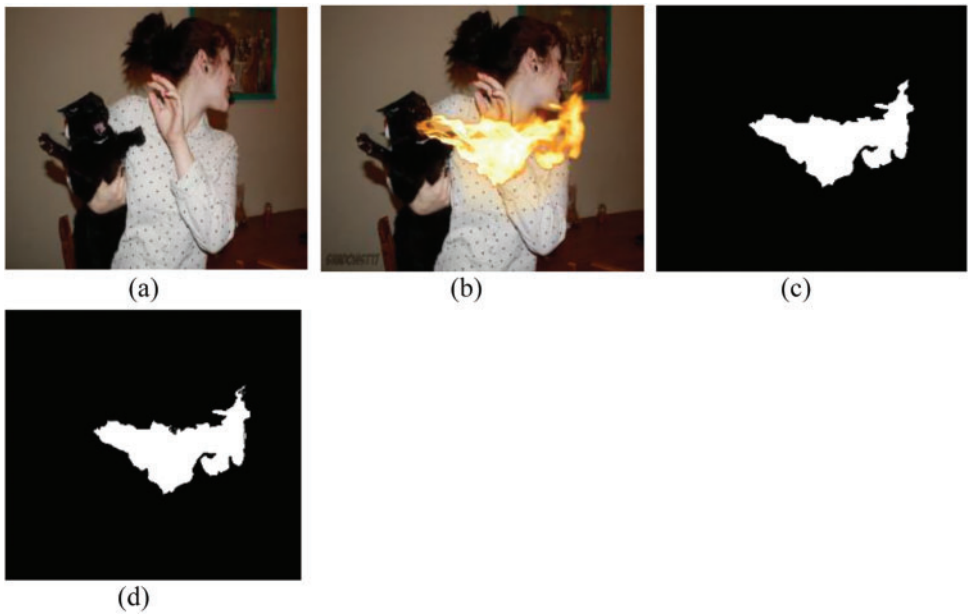

Figure 12: The sample images of IMD2020 dataset. (a) Original image (b) Manipulated image (c) GT and (d) Output

The summary of the different hyper parameters is depicted in Table 6. We experimented with batch sizes 8, 16, 32, 64, and 128 models, and the highest performance was achieved with a batch size of 64. For lower batch size models, training time also increased. At the same time, performance was also less. Furthermore, we selected 8, 16, and 32 heads in the ViT encoder for global attention. However, with the 8 heads model achieved better performance. Moreover, lower embedding dimensions resulted in lower performance, and higher dimensions increased computation costs.

Table 6: Different hyper parameters used in the study

Hyper parameters	Value	Details
Batch size	64	Used in training
No of heads	8	Assigned in the ViT encoder
Embedding dimension	512	Assigned in the ViT encoder
Window size	7×7	Used in W-MSA

(Continued)

Table 6 (continued)

Hyper parameters	Value	Details
Focusing parameter	2.0	Used in binary focal loss
Trade-off value	0.73	Used in binary focal loss

5 Limitation and Conclusion

Digital platforms are widely used for sharing images. Due to the large-scale availability of advanced image editing tools, tampered images are generated. The tampered images potentially cause serious consequences. Several manual techniques have been utilized to detect forged images, which is time-consuming and requires forensic experts. Recently, ML and DL have been successfully used to automate tampered image localization. The ML-based method requires hand-crafted features for the model training. Therefore, the probability of error is associated with these techniques. On the other hand, the based model improved the localization of tampered regions by automatically extracting the spatial features. However, DL-based methods require extensive data to train the model. In addition, the shallow CNN-based encoder and decoder model lacks the global co-relation of the features. Therefore, a CNN-based encoder with spatial attention using a transformer has been developed. The CNN encoder extracts local high-dimension features and global co-relation of the spatial features provided by the transformer. Our decoder is a purely CNN-based module, which upsamples the features to localized tampered mask images.

We validated the model on three standard datasets and compared it with six different methods. The FLTNet achieved an F1-score of 0.80, 0.84, and 0.86 on the CASIA V1, CASIA V2, and CoMoFoD datasets. In addition, the loss of the model on the three datasets is much less compared to the other methods. Furthermore, the AUC value of the model is 87%, 92% and 95%, respectively. The FLTNet performance needs to be tested on diverse real-time datasets. In addition, the computation costs of the algorithm need further reduction. Future studies will explore transformer-based lightweight encoders and decoders for tampered image localization. In addition, an explainable AI module can be implemented for better decision-making that will build more trust in the model.

Acknowledgement: We express our sincere thanks to the Department of Computer Engineering and Applications, GLA University, Mathura for providing support to conduct the research.

Funding Statement: This research received no external funding.

Author Contributions: The authors confirm contribution to the paper as follows: Study conception and design: Sachin Sharma; data collection: Sachin Sharma, Brajesh Kumar Singh; analysis and interpretation of results: Hitendra Garg, Sachin Sharma; draft manuscript preparation: Brajesh Kumar Singh; supervision: Hitendra Garg, Brajesh Kumar Singh. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The dataset used in the study can be downloaded from the following open access repository: https://www.vcl.fer.hr/comofod/ (accessed on 21 January 2025).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

Abbreviations

RGB	Red Blue Green
IFDL	Image Forgery Detection and Localization
JPEG	Joint Photographic Experts Group
PRNU	Photo Response Non-Uniformity

References

1. Bhardwaj A, Bharany S, Osman Ibrahim A, Almogren A, Ur Rehman A, Hamam H. Unmasking vulnerabilities by a pioneering approach to securing smart IoT cameras through threat surface analysis and dynamic metrics. Egypt Inform J. 2024;27(7):100513. doi:10.1016/j.eij.2024.100513.
2. Tyagi S, Yadav D. A detailed analysis of image and video forgery detection techniques. Vis Comput. 2023;39(3):813–33. doi:10.1007/s00371-021-02347-4.
3. Shen C, Kasra M, Pan W, Bassett GA, Malloch Y, O'Brien JF. Fake images: the effects of source, intermediary, and digital media literacy on contextual assessment of image credibility online. New Medium Soc. 2019;21(2):438–63. doi:10.1177/1461444818799526.
4. Shukla DK, Bansal A, Singh P. A survey on digital image forensic methods based on blind forgery detection. Multimed Tools Appl. 2024;83(26):67871–902. doi:10.1007/s11042-023-18090-y.
5. Kaur G, Singh N, Kumar M. Image forgery techniques: a review. Artif Intell Rev. 2023;56(2):1577–625. doi:10.1007/s10462-022-10211-7.
6. Yang F, Li J, Lu W, Weng J. Copy-move forgery detection based on hybrid features. Eng Appl Artif Intell. 2017;59(11):73–83. doi:10.1016/j.engappai.2016.12.022.
7. Warif NBA, Idris MYI, Wahab AWA, Salleh R. An evaluation of Error Level Analysis in image forensics. In: 2015 5th IEEE International Conference on System Engineering and Technology (ICSET); 2015 Aug 10–11; Shah Alam, Malaysia: IEEE; 2015. p. 23–8. doi:10.1109/ICSEngT.2015.7412439.
8. Korus P. Digital image integrity-a survey of protection and verification techniques. Digit Signal Process. 2017;71(4):1–26. doi:10.1016/j.dsp.2017.08.009.
9. Bourouis S, Alroobaea R, Alharbi AM, Andejany M, Rubaiee S. Recent advances in digital multimedia tampering detection for forensics analysis. Symmetry. 2020;12(11):1811. doi:10.3390/sym12111811.
10. Siegel D, Kraetzer C, Seidlitz S, Dittmann J. Media forensics considerations on DeepFake detection with hand-crafted features. J Imaging. 2021;7(7):108. doi:10.3390/jimaging7070108.
11. Zhuang P, Li H, Tan S, Li B, Huang J. Image tampering localization using a dense fully convolutional network. IEEE Trans Inf Forensics Secur. 2021;16:2986–99. doi:10.1109/TIFS.2021.3070444.
12. Yadav DP, Sharma B, Chauhan S, Ben Dhaou I. Bridging convolutional neural networks and transformers for efficient crack detection in concrete building structures. Sensors. 2024;24(13):4257. doi:10.3390/s24134257.
13. Li Y, Cao J, Xu Y, Zhu L, Dong ZY. Deep learning based on Transformer architecture for power system short-term voltage stability assessment with class imbalance. Renew Sustain Energy Rev. 2024;189(4):113913. doi:10.1016/j.rser.2023.113913.
14. Cozzolino D, Gragnaniello D, Verdoliva L. A novel framework for image forgery localization. arXiv:1311.6932. 2013.
15. Yarlagadda SK, Güera D, Bestagini P, Zhu FM, Tubaro S, Delp EJ. Satellite image forgery detection and localization using GAN and one-class classifier. arXiv:1802.04881. 2018.
16. Kumar A, Bhavsar A. Copy-move forgery classification via unsupervised domain adaptation. arXiv:1911.07932. 2019.
17. Le N, Retraint F. An improved algorithm for digital image authentication and forgery localization using demosaicing artifacts. IEEE Access. 2019;7:125038–53. doi:10.1109/ACCESS.2019.2938467.
18. Bi X, Zhang Z, Xiao B. Reality transforms adversarial generators for image splicing forgery detection and localization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision; 2021; Montreal, Canada: IEEE. p. 14294–303. doi:10.1109/ICCV48922.2021.01403.
19. Cozzolino D, Verdoliva L. Camera-based image forgery localization using convolutional neural networks. In: 2018 26th European Signal Processing Conference (EUSIPCO); 2018 Sep 3–7; Rome, Italy: IEEE; 2018. p. 1372–6.

20. Bunk J, Bappy JH, Mohammed TM, Nataraj L, Flenner A, Manjunath BS, et al. Detection and localization of image forgeries using resampling features and deep learning. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW); 2017 Jul 21–26; Honolulu, HI, USA: IEEE; 2017. p. 1881–9. doi:10.1109/CVPRW.2017.235.
21. Guillaro F, Cozzolino D, Sud A, Dufour N, Verdoliva L. TruFor: leveraging all-round clues for trustworthy image forgery detection and localization. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2023 Jun 17–24; Vancouver, BC, Canada: IEEE; 2023. p. 20606–15. doi:10.1109/CVPR52729.2023.01974.
22. Guo X, Liu X, Ren Z, Grosz S, Masi I, Liu X. Hierarchical fine-grained image forgery detection and localization. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2023 Jun 17–24; Vancouver, BC, Canada: IEEE; 2023. p. 3155–65. doi:10.1109/CVPR52729.2023.00308.
23. Hao J, Zhang Z, Yang S, Xie D, Pu S. TransForensics: image forgery localization with dense self-attention. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV); 2021 Oct 10–17; Montreal, QC, Canada: IEEE; 2021. p. 15035–44. doi:10.1109/ICCV48922.2021.01478.
24. Bianchi T, Piva A. Image forgery localization via block-grained analysis of JPEG artifacts. IEEE Trans Inf Forensics Secur. 2012;7(3):1003–17. doi:10.1109/TIFS.2012.2187516.
25. Ferrara P, Bianchi T, De Rosa A, Piva A. Image forgery localization via fine-grained analysis of CFA artifacts. IEEE Trans Inf Forensics Secur. 2012;7(5):1566–77. doi:10.1109/TIFS.2012.2202227.
26. Li H, Luo W, Qiu X, Huang J. Image forgery localization via integrating tampering possibility maps. IEEE Trans Inf Forensics Secur. 2017;12(5):1240–52. doi:10.1109/TIFS.2017.2656823.
27. Singh B, Sharma DK. Image forgery over social media platforms-A deep learning approach for its detection and localization. In: 2021 8th International Conference on Computing for Sustainable Global Development (INDIACom); 2021; Bharati Vidyapeeth, New Delhi, India: Springer. p. 705–9.
28. Dua S, Singh J, Parthasarathy H. Detection and localization of forgery using statistics of DCT and Fourier components. Signal Process Image Commun. 2020;82(1):115778. doi:10.1016/j.image.2020.115778.
29. Rao Y, Ni J, Xie H. Multi-semantic CRF-based attention model for image forgery detection and localization. Signal Process. 2021;183(5):108051. doi:10.1016/j.sigpro.2021.108051.
30. Jabeen S, Khan UG, Iqbal R, Mukherjee M, Lloret J. A deep multimodal system for provenance filtering with universal forgery detection and localization. Multimed Tools Appl. 2021;80(11):17025–44. doi:10.1007/s11042-020-09623-w.
31. Liu B, Pun CM. Deep fusion network for splicing forgery localization. In: Proceedings of the European Conference On Computer Vision (ECCV) Workshops; 2018; Munich, Germany: Springer. p. 237–51. doi:10.1007/978-3-030-11012-3_21.
32. Lin X, Li CT. PRNU-based content forgery localization augmented with image segmentation. IEEE Access. 2020;8:222645–59. doi:10.1109/ACCESS.2020.3042780.
33. Rao Y, Ni J. Self-supervised Doma in adaptation for forgery localization of JPEG compressed images. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV); 2021 Oct 10–17; Montreal, QC, Canada: IEEE; 2021. p. 15014–23. doi:10.1109/ICCV48922.2021.01476.
34. Kumar V, Kansal V, Gaur M. Multiple forgery detection in video using convolution neural network. Comput Mater Contin. 2022;73(1):1347–64. doi:10.32604/cmc.2022.023545.
35. Bodapati JD, Sajja R, Naralasetti V. An efficient approach for semantic segmentation of salt domes in seismic images using improved UNET architecture. J Inst Eng Ind Ser B. 2023;104(3):569–78. doi:10.1007/s40031-023-00875-2.
36. Cao H, Wang Y, Chen J, Jiang D, Zhang X, Tian Q, et al. Unet-like pure transformer for medical image segmentation. In: European Conference on Computer Vision; 2022; Cham: Springer Nature Switzerland. p. 205–18. doi:10.1007/978-3-031-25066-8_9.
37. Dong J, Wang W, Tan T. CASIA image tampering detection evaluation database. In: 2013 IEEE China Summit and International Conference on Signal and Information Processing; 2013 Jul 6–10; Beijing, China: IEEE; 2013. p. 422–6. doi:10.1109/ChinaSIP.2013.6625374.

38. Wu Y, Abd-Almageed W, Natarajan P. Busternet: detecting copy-move image forgery with source/target localization. In: Proceedings of the European Conference on Computer Vision (ECCV); 2018; Munich, Germany: Springer. p. 168–84. doi:10.1007/978-3-030-01231-1_11.
39. Tralic D, Zupancic I, Grgic S, Grgic M. CoMoFoD—New database for copy-move forgery detection. In: 55th International Symposium ELMAR-2013; 2013; Zadar, Croatia. p. 49–54.
40. Badrinarayanan V, Kendall A, Cipolla R. SegNet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans Pattern Anal Mach Intell. 2017;39(12):2481–95. doi:10.1109/TPAMI.2016.2644615.
41. Ronneberger O, Fischer P, Brox T. U-Net: convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015; 2015; Cham: Springer International Publishing. p. 234–41. doi:10.1007/978-3-319-24574-4_28.
42. Wu Y, AbdAlmageed W, Natarajan P. ManTra-net: manipulation tracing network for detection and localization of image forgeries with anomalous features. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2019 Jun 15–20; Long Beach, CA, USA: IEEE; 2019. p. 9535–44. doi:10.1109/cvpr.2019.00977.
43. Bappy JH, Simons C, Nataraj L, Manjunath BS, Roy-Chowdhury AK. Hybrid LSTM and encoder-decoder architecture for detection of image forgeries. IEEE Trans Image Process. 2019;28(7):3286–300. doi:10.1109/TIP.2019.2895466.
44. Zhou P, Han X, Morariu VI, Davis LS. Learning rich features for image manipulation detection. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2018 Jun 18–23; Salt Lake City, UT, USA: IEEE; 2018. p. 1053–61. doi:10.1109/CVPR.2018.00116.
45. Hu X, Zhang Z, Jiang Z, Chaudhuri S, Yang Z, Nevatia R. SPAN: Spatial pyramid attention network for image manipulation localization. In: Computer Vision-ECCV 2020: 16th European Conference; 2020; Glasgow, UK. p. 312–28.
46. Novozamsky A, Mahdian B, Saic S. IMD2020: a large-scale annotated dataset tailored for detecting manipulated images. In: 2020 IEEE Winter Applications of Computer Vision Workshops (WACVW); 2020 Mar 1–5; Snowmass Village, CO, USA: IEEE; 2020. p. 71–80. doi:10.1109/wacvw50321.2020.9096940.