Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Active Image Manipulation Detection and Localization using TransUNet for Fragile Watermarking

Zohaib Hamdule

24cse2014@nitgoa.ac.in

National Institute of Technology Goa https://orcid.org/0009-0007-7490-225X

Venkatanareshbabu Kuppili

National Institute of Technology Goa

Research Article

Keywords: Active Image Manipulation Detection, Fragile Watermarking, Image Forensics, TransUNet, Tamper Localization

Posted Date: July 28th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10474615/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Active Image Manipulation Detection and Localization using TransUNet for Fragile Watermarking

Mohammad Zohaib Hamdule[1*†] and Venkatanareshbabu Kuppili[1†]

[1*]Department of Computer Science and Engineering, National Institute of Technology Goa, Cuncolim, 403703, Goa, India.

*Corresponding author(s). E-mail(s): 24cse2014@nitgoa.ac.in;
Contributing authors: venkatanaresh@nitgoa.ac.in;
†These authors contributed equally to this work.

Abstract

Purpose: The widespread availability of advanced photo editing tools has created a critical need for robust digital image forensics. Active Image Manipulation Detection and Localization (AIMDL) offers highly accurate detection and precise spatial localization compared to passive forensic methods. However, AIMDL faces an inherent trade-off between watermark transparency (invisibility) and robustness against routine transformations. **Methods:** This paper proposes a novel AIMDL framework that utilizes TransUNet architectures for the generation and recovery of watermarks, coupled with a multi-head UNet-based Forensic Module to handle image authentication, tampering detection, and spatial localization. The training pipeline integrates Benign Noise and Manipulation modules alongside a carefully curated set of loss functions to optimize watermark quality and forensic accuracy. Models were trained at different watermark scaling factors (0.10 and 0.05), the framework allows users to dynamically balance forensic performance and perceptual quality at test time. **Results:** Experimental results confirm that the proposed pipeline achieves high authentication and detection accuracy (> 0.95) and precise pixel-level localization (IoU > 0.85) of malicious edits while remaining robust against benign transformations like compression and scaling. **Conclusion:** The framework offers reliable active image forensics, allowing users to tune the balance between imperceptibility and forensic strength according to application-specific security demands.

Keywords: Active Image Manipulation Detection, Fragile Watermarking, Image Forensics, TransUNet, Tamper Localization

1 Introduction

The widespread availability and power of modern photo editing tools have made it significantly easier to alter digital images [1], such as by removing objects or changing background details, which highlights the critical need for reliable methods to ensure image authenticity. Image Manipulation and Detection and Localization (IMDL) deals with identifying an image as manipulated and localizing the manipulations introduced in the image. These solutions are primarily categorized into passive and active approaches [2]. Passive methods, often called blind forensics, analyze the inherent statistical properties and forensic artifacts such as lighting inconsistencies, noise patterns, or compression irregularities left within an image to detect tampering without prior knowledge of its original state. In contrast, active methods rely on pre-embedded authentication data, such as fragile watermarks or digital signatures, introduced at the time of image creation; any subsequent modification disrupts this embedded information, allowing the system to precisely pinpoint and localize tampered regions. Active methods offer several advantages over passive techniques. Primarily, they offer significantly more accurate detection and precise localization of manipulations [3]. Since, active methods involve embedding a signal at the time of creation, it inherently provides authentication capabilities, determining whether an image comes from the original source. Active methods identify and localize manipulations made after the watermark was added, and any manipulation made before the watermarking is not tracked. Images are thus expected to be watermarked at the time of creation or before distribution. Active IMDL also comes with a set of challenges. The first major challenge is that embedding a digital watermark inherently alters the original pixel values, which can degrade the visual quality of the image. This degradation must remain imperceptible to the human eye, requiring sophisticated embedding techniques. Focusing on transparency of the watermark makes it too fragile, which means benign or routine transformations like JPEG compression can corrupt or completely destroy the embedded watermark. Because transparency and robustness are inherently antagonistic [4], optimizing one inevitably compromises the other, increasing invisibility reduces the signal's resilience to compression, while boosting robustness introduces visible, unwanted artifacts. Therefore, any practical implementation of Active IMDL must allow users to balance and trade-off between robustness and transparency based on the specific security and quality demands. This study briefly summarizes recent works in Active IMDL and proposes a novel Active IMDL pipeline utilizing TransUNets [5] and a set of carefully curated loss functions to optimize image quality, robustness, localization performance.

2 Performance Metrics

Before delving into the survey of existing Active IMDL techniques, it is imperative to understand how the performance of these techniques is evaluated. Performance metrics are categorized into three groups based on the aspect they evaluate: Classification (Authentication/Detection) Metrics, Localization Metrics, and Transparency Metrics.

2.1 Classification Metrics

These metrics evaluate the system's ability to correctly classify an image as watermarked/non-watermarked (authentication) and tampered/untampered (detection). Let TP be True Positives (correctly identified forgeries), TN be True Negatives (correctly identified authentic images), FP be False Positives (authentic images falsely flagged as forged), and FN be False Negatives (forgeries missed by the system).

- **Accuracy:** The proportion of total images correctly identified as either authentic or tampered out of all tested images.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

- **Precision:** The ratio of correctly identified forged images to the total number of images flagged as forged, measuring the system's resistance to false alarms.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

- **Recall:** The ratio of correctly identified forged images to the actual number of images, measuring the system's ability to catch all instances of tampering.

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

- **F1-Score:** The harmonic mean of precision and recall, providing a single, balanced metric for overall classification performance, especially useful if the dataset has an unequal number of authentic and forged images.

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4}$$

2.2 Localization Metrics

These metrics assess the system's precision in pinpointing the exact spatial boundaries and regions where manipulation occurred within an image. Let P represent the set of pixels predicted as manipulated by the network, and Y represent the ground-truth set of truly manipulated pixels.

- **Pixel-Wise Accuracy:** The percentage of individual pixels correctly classified as either "tampered" or "untampered" across the entire image grid.

$$Pixel\text{-}Wise\ Accuracy = \frac{TP_{pixel} + TN_{pixel}}{TP_{pixel} + TN_{pixel} + FP_{pixel} + FN_{pixel}} \tag{5}$$

- **Intersection over Union (IoU):** A ratio that measures the overlap between the predicted manipulated area and the actual ground-truth manipulated area, divided

by the total area of their union.

$$IoU = \frac{|P \cap Y|}{|P \cup Y|} \tag{6}$$

- **F1-Score (Pixel-Level):** Evaluates the pixel-level balance between precision and recall, determining how accurately the system maps the shape and edges of the forged region without missing tampered pixels or falsely penalizing authentic ones.

$$F1\text{-}Score_{pixel} = \frac{2 \times |P \cap Y|}{|P| + |Y|} \tag{7}$$

2.3 Transparency Metrics

These metrics quantify the visual quality and imperceptibility of the embedded watermark, measuring how much distortion the active technique introduces into the host image. Let I be the original image of size $M \times N$, K be the watermarked image, and MAX_I be the maximum possible pixel value of the image (typically 255 for 8-bit depth images).

- **Mean Square Error (MSE):** Measures the average squared intensity difference between the pixels of the original cover image and the watermarked image; lower values signify less distortion.

$$MSE = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [I(i,j) - K(i,j)]^2 \tag{8}$$

- **Peak Signal-to-Noise Ratio (PSNR):** An expression of the ratio between the maximum possible power of the signal and the corrupting noise (distortion) introduced by watermarking; higher decibel (dB) values indicate better image quality.

$$PSNR = 10 \cdot \log_{10} \left(\frac{MAX_I^2}{MSE} \right) \tag{9}$$

- **Structural Similarity Index (SSIM):** A perception-based metric that evaluates changes in structural information, luminance, and contrast, measuring how closely the human visual system perceives the watermarked image relative to the original. It evaluates structural degradation based on luminance (μ), contrast (σ), and structural cross-covariance (σ_{IK}). Let c_1 and c_2 be stabilizing constants.

$$SSIM(I,K) = \frac{(2\mu_I\mu_K + c_1)(2\sigma_{IK} + c_2)}{(\mu_I^2 + \mu_K^2 + c_1)(\sigma_I^2 + \sigma_K^2 + c_2)} \tag{10}$$

3 Literature Review

3.1 Traditional Spatial and Transform Domain Watermarking

As sophisticated digital editing tools make it increasingly easy to manipulate digital images, the field of image forensics has shifted from passive detection, which looks for editing artifacts after the fact, to active defense strategies that embed protective signals directly into the original media. Early active methods relied on traditional mathematical algorithms in the spatial or frequency domains. These traditional approaches are highly deterministic and computationally efficient. For instance, Sahu [6] proposed a blind, strictly fragile approach that embeds chaotic, logistic map based fragile watermarks into the image's pixel data to provide highly sensitive tampering detection without relying on machine learning. Specifically, it generates watermark bits using the sensitivity of the logistic map and embeds them into the least significant bits via a logical XOR operation with intermediate significant bits, ensuring any structural alteration disrupts the underlying logic. the method achieves excellent perceptual transparency (PSNR of 51.14 dB) and accurate spatial tamper localization.

To achieve both tamper localization and content recovery, researchers like Abdelhakim et al. [7] utilized Schur decomposition [8] and the Discrete Wavelet Transform (DWT) [9], combined with K-means clustering [10], while Lefèvre et al. [11] utilized error control codes to distinguish between harmless edits and malicious tampering. In [7], when the system extracts the watermark and detects faulty pixels, it uses K-means to group these errors based on their physical location. This successfully separates large, concentrated clusters of actual, malicious tampering (like someone pasting an object into the photo) from random, isolated pixel errors caused by normal digital noise, allowing the system to accurately map the fake region's boundaries. Conversely, Lefèvre et al. in [11] globally distribute redundant error control bits across intact frequency regions, allowing the algorithm to computationally reconstruct destroyed data if a localized attack occurs.

In highly sensitive fields like radiology, where even slight pixel distortions due to watermarking are unacceptable, Bouarroudj et al. [12] introduced strictly reversible watermarking frameworks that guarantee a medical image can be perfectly restored to its exact original state with zero data loss after authentication and localization. The "zero-distortion mechanism" approach mathematically records exactly how the image pixels are altered when the security watermark is initially hidden inside them. Once a hospital system verifies the watermark to ensure the image hasn't been hacked, the algorithm uses this record to run in reverse to completely undo the distortions caused by watermarking.

3.2 Deep Learing Approaches

While traditional mathematical watermarks are effective, they are often unintentionally destroyed by the complex transformations of modern generative models, prompting the development of proactive, deep learning-based defenses. Instead of simply embedding a passive mark, these systems actively optimize neural "templates" or perturbations that interact destructively with attacking generative models. Asnani

et al. introduced Proactive Image Manipulation Detection (PIMD) [13], which adds an imperceptible template to an image; if a generative model alters the image, the template's structure degrades, allowing the system to easily flag the manipulation. Detection is performed by extracting the latent template via an encoder network and calculating its cosine similarity against the original orthogonal template set, where a degraded similarity score definitively flags manipulation.

Expanding on this, newer architectures like Petrub, Attend, Detect, and Localize (PADL) [14] by Bartolucci et al. use cross-attention mechanisms to map these degraded perturbations directly back to the specific pixels that were altered, providing highly accurate spatial localization against advanced diffusion models. The framework's encoder generates an image-specific perturbations, while the decoder analyzes the structural degradation of these perturbations to accurately map the spatial boundaries of the manipulation. To ensure these neural watermarks survive routine internet transmission, like standard JPEG compression, while fracturing under malicious edits, Zhao et al. [15] introduced a deep semi-fragile watermark embedded as a transparent latent feature mask. Through adversarial training, the mask is mathematically enforced to retain its integrity under lossy compression and resizing, while instantly fracturing upon localized semantic tampering (e.g., face-swapping or splicing), yielding state-of-the-art forgery localization heatmaps. Additionally, models like DeepMark [16], proposed by Abrar and Sheikh, utilizes flow-based invertible neural networks, which allow the forward (embedding) and reverse (extraction) processes to share the exact same weights. This bijective architecture mathematically eliminates the embedding of redundant features, maximizing capacity and visual invisibility.

A major structural bottleneck for these deep learning defenses is their inability to scale to high resolution media due to GPU memory constraints. Duszejko and Piotrowski resolved this with 4Ksecure [17], a universal method capable of actively protecting images of virtually any resolution. By utilizing an intermediate resolution downsampling protocol alongside two cooperating neural networks, 4KSecure generates a resolution independent signature mask that surpasses previous scalability limitations while maintaining high perceptual transparency. Furthermore, to provide transparent reasoning for why an image is flagged as manipulated, Wang et al. introduced FractalForensics [18], which utilizes parameter-driven fractal watermarks combined with an "entry-to-patch" mapping embedding strategy. If a specific facial region is manipulated, the underlying geometry of the fractal in that exact patch is broken, allowing the system to explicitly highlight the distorted geometry to the user. This technique generates the watermarks using user selected chaotic encryption keys and distributes the recursive geometry implicitly into discrete patches, guaranteeing explainability when the decoding module attempts to reconstruct the missing components. Table 1 summarizes the structural progression of these active image manipulation detection methodologies.

Table 1 Summary of recent works in Active Image Manipulation Detection and Localization

Year	Work	Ref.	Core Idea	Type	Robust	Task
2025	**4KSecure**	[17]	NN signatures for images of virtually any resolution.	DL	Yes (Semi)	Detection, Localization
2025	**DeepMark**	[16]	Flow-based invertible NN for watermarking image and video.	DL	Yes	Authentication, Detection, Localization
2025	**Medical Reversible**	[12]	Reversible fragile watermarking for medical image authentication.	Trad	No	Authentication, Localization
2025	**FractalForensics**	[18]	Fractal watermarks for proactive deepfake detection and localization.	DL	Yes	Detection, Localization
2024	**Deep Semi-fragile**	[15]	Learned semi-fragile watermarks robust to benign distortions.	DL	Yes	Detection, Localization
2024	**PADL**	[14]	Image-conditioned perturbations using cross-attention for robust defense.	DL	Yes	Detection, Localization
2022	**PIMD**	[13]	Templates added to images to improve manipulation discrimination.	DL	Yes	Detection, Localization
2021	**Logistic Map**	[6]	Chaotic, logistic map based fragile watermarking for tamper detection	Trad	No	Detection, Localization
2021	**ECC-SF**	[11]	Semi-fragile watermarking using error control codes.	Trad	Yes	Localization
2019	**K-means**	[7]	K-means fragile watermarking with content recovery capability.	Trad	No	Detection, Localization, Recovery

4 Proposed Methodology

A review of active image forensics highlights the need for a system that performs all three core tasks: authentication, detection, and localization. Existing schemes often perform direct, black-box localization on manipulated images, making it unclear whether decisions rely on watermark integrity or blind image artifacts. To solve this, our proposed model first recovers the watermark and uses it for downstream forensic tasks. Additionally, the pipeline injects manipulations prior to watermarking during

training. This guarantees that only post-watermarking tampering is tracked and also prevents the model from blindly learning the distribution of the manipulation module.

In the proposed system, TransUNets [5] are utilized to generate and recover watermarks. A Benign Noise Module introduces non-malicious transformations such as compression, brightness, and contrast adjustments to improve the watermark's robustness against standard image processing. Conversely, the Manipulation Module applies malicious manipulations such as splicing, copy-move, random erasing, that the system must detect and localize if they occur after watermarking. Once the watermark is recovered, it is passed to a UNet-based [19] Multi-Head Forensic Module. This module authenticates the image by verifying the watermark's presence and simultaneously detects and localizes any post-watermark manipulations. A collection of loss functions, discussed later, ensures watermark quality, invisibility, and high performance across both watermark recovery and forensic tasks.

Figure 1 illustrates the overall Active IMDL pipeline, and Figure 2 details the specifications of the Benign Noise and Manipulation modules.

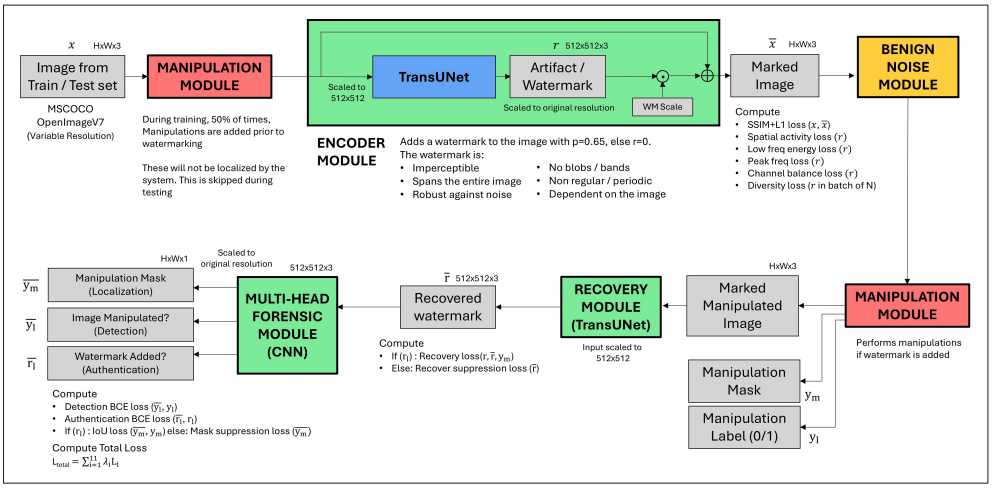


Fig. 1 Overall Proposed Active Image Manipulation Detection and Localization Pipeline

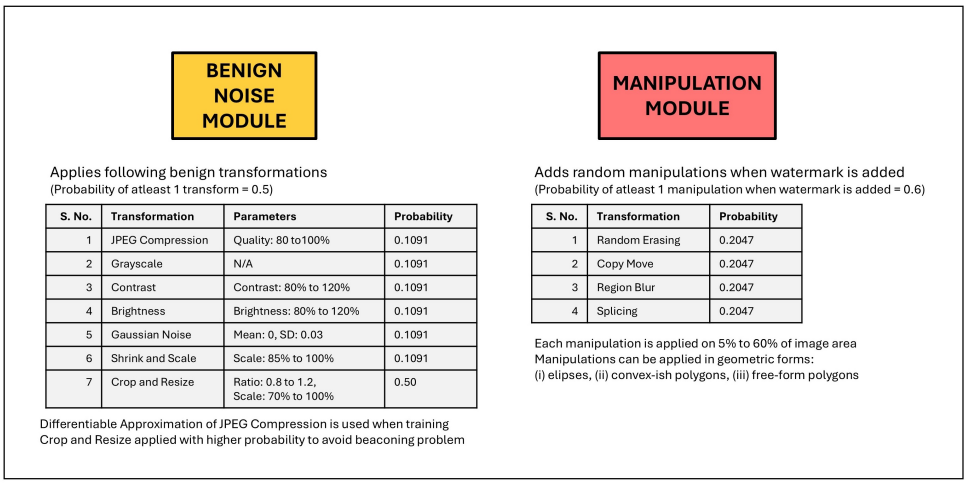


Fig. 2 Specification of Benign Noise Module and Manipulation Module

As shown in Figure 1, during training, manipulations are also added before the watermark is added. This is done to ensure two things: (i) The system learns to only track and localize manipulations added after the watermark is added, everything before that is considered original state of the image. (ii) The forensic module does not ignore the watermark completely and start detecting and localizing manipulations independent of the watermark.

In both the Benign Noise and Manipulation modules, transformation probabilities follow a binomial distribution. When applied independently, the probability of at least one transformation occurring is set to 0.5 for benign noise and 0.6 for malicious manipulations (further details regarding task-specific probabilities are discussed in section 4.3). The notable exception is the Crop and Resize transformation within the Benign Noise Module, which is applied independently with a significantly higher probability (0.5). This elevated rate mitigates an artifacting issue observed during preliminary testing, where blobs and bands would appear at edges of watermarked images and act as a "beacon", allowing the recovery module to trivially detect the watermark's presence.

TransUNets were used to implement Encoder Module (generating and embedding watermarks) and Recovery Module (recovering watermark from marked images). TransUNet were preferred due to their ability to combine the strengths of CNNs (ability to extract spatial features) and and Transformers [20] (capturing global dependencies) [5]. Figure 3 shows the architecture of the TransUNet used.

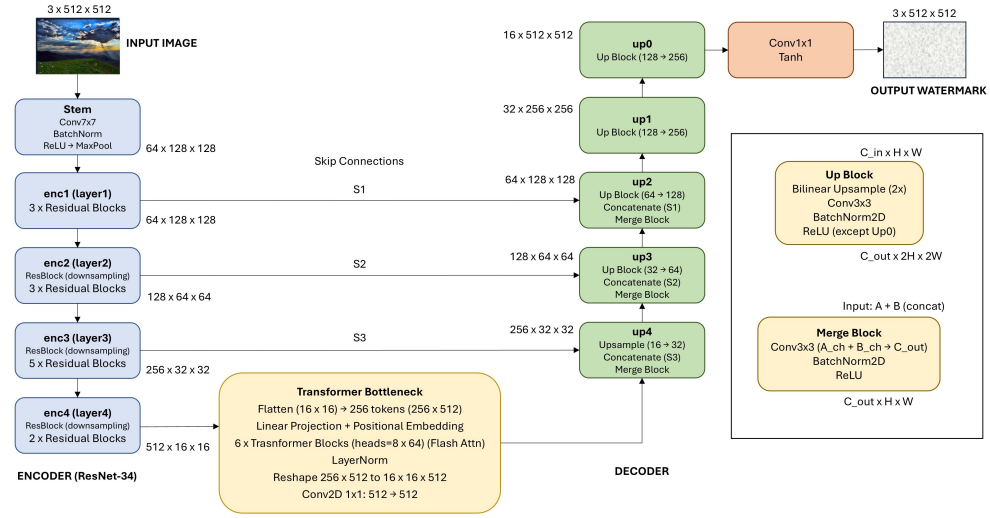


Fig. 3 Architecture of TransUNet used for generating and recovering watermarks.

4.1 Dataset Details

Images were fetched from MSCOCO [21] and OpenImageV7 [22] datasets using Fifty-One Dataset Zoo API [23], images larger than 1024 pixels in any dimension or having max aspect ratio of 1.6 were filtered. Approximately twice as many images were fetched from MSCOCO than OpenImageV7. A total of 3000 images were obtained, and were split into 2100 for training, 300 for validation and 600 for testing. Since splicing manipulation requires patches from a donor image, a patch set of 1000 images was extracted from the training set.

4.2 Loss Functions

Loss functions optimized during training can be broadly classified into 4 groups. (i) Encoder Module losses ensure original and watermarked images are structurally and visually similar. (ii) Watermark Quality losses ensure watermark is spread across all channels, spread across spatial and frequency domains, does not contain blobs and bands, and is distinct for each image. (iii) Recovery Module losses ensure watermark recovered is similar to the embedded watermark. (iv) Finally, Forensic Module losses are used for the supervision of authentication, detection and localization tasks. Details of the loss functions used is given in Table 2.

Peak frequency loss was assigned a lower weight due to its high magnitude, this prevents it from overwhelming the total loss function and implicitly muting other losses. Recovery suppression loss was also assigned a relatively higher weight as it was found to correlate directly with the downstream authentication task. Authentication

Table 2 Summary of loss functions, their purpose and weight in total loss

#	Loss Function	Inputs	Description	Purpose	Range	Practical Range	Initial Weight
1	SSIM+MAE Loss	Original Image, WMed Image	Weighted sum of SSIM and MAE. (0.95 * SSIM-Loss + 0.05 * MAELoss)	Structural and visual similarity	$[0, 1]$	$[0, 1]$	1.5
2	Spatial Activity Loss	Watermark, Patch size	Mean of ratio of local energy / global energy, squashed using e^{-kx}	Ensures all patches contribute to WM. No cold spots.	$[0, 1]$	$[0, 1]$	0.5
3	Low Frequency Energy Loss	Watermark, Inner radius, Outer radius	Ratio of energy in mid frequencies / total energy. Blobs of size $\in [D/2, D/8]$ penalized	Prevents blobs and bands in the WM.	$[0, 1]$	$[0, 1]$	0.5
4	Peak Frequency Loss	Watermark	Ratio of peak energy / mean energy in FFT, squashed using $\ln(1 + x)$	Prevents periodicity in watermark	$[0, \infty)$	< 10	0.05
5	Diversity Loss	Watermark, WM label	Mean of pairwise cosine similarity between all WM in the batch	Encourages unique, distinct watermarks	$[-1, 1]$	$[-1, 1]$	1.5
6	Channel Balance Loss	Watermark	Mean of squared Coef. Of Variation between, each channel energy and mean energy	Prevents watermark to be concentrated in one channel	$[0, 1]$	$[0, 1]$	1.5
7	Recovery Loss	WM, Recovered WM, Manipulation mask	Weighted sum of: (1) NCC loss [24], (2) L1 loss between actual and recovered WMs, (3) L1 loss between log of FFTs, all computed over untampered areas. ($L_1 + 0.5 * L_2 + 0.3 * L_3$)	Similarity between WM and recovered WM (in spatial and frequency domains). Avoids blobs in recovered WM.	$[0, \infty)$	< 1	5
8	Recovery Suppression Loss	WM Pred., WM Label	L1 loss between predicted WM and all 0s, if WM is not added.	Suppresses predicted WM when no WM is added.	$[0, 1]$	$[0, 1]$	5
9	Detection Loss	Manip Label, Manip Pred.	Binary Cross Entropy Loss	Improves detection performance	$[0, \infty)$	< 1	2
10	Authentication Loss	WM Label, WM Pred.	Binary Cross Entropy Loss	Improves authentication performance	$[0, \infty)$	< 1	4
11	IoU Loss	Manip Mask, Manip Mask Pred.	Intersection over Union Loss	Improves localization performance	$[0, 1]$	$[0, 1]$	3
12	Mask Suppression Loss	Manip Mask Pred., WM Label	Binary Cross Entropy Loss between Pred. Mask (logits) and all 0s.	Suppresses predicted mask when no WM is added.	$[0, \infty)$	< 1	0.5

Original Image, WMed Image $\in [0, 1]^{\{512\times512\times3\}}$
Watermark, Recovered WM $\in [-1, 1]^{\{512\times512\times3\}}$
Manipulation Mask $\in \{0, 1\}^{\{512\times512\times1\}}$ (binary masks)
Pred. Manip Mask $\in \mathbb{R}^{\{512\times512\times1\}}$ (logits, before sigmoid)

Legends:
- **Red:** ENCODER Module losses
- **Blue:** Watermark Quality losses
- **Yellow/Orange:** RECOVERY Module losses
- **Green:** FORENSIC Module losses

loss was itself assigned higher weight than other forensic tasks due to the 65:35 imbalance in watermark / non-watermarked images (discussed in section 4.3) and due to the fact that the model was struggling with authentication in preliminary test runs.

4.3 Training and Testing

For the Forensic Module to have optimal performance, it needs to have a good balance of images across all tasks. The authentication head needs an even balance of watermarked and non watermarked images. The detection head needs an even distribution of manipulated and unmanipulated images that are watermarked. And, the localization head needs a large amount of manipulated, watermarked images. To ensure an even distribution across all tasks and cases, the system follows a probability distribution shown in Table 3.

Table 3 Joint Probability Distribution for Manipulation applied and Watermark added

	Watermark (W)	no Watermark ($\neg W$)	Total
Manipulation (M)	0.39	0	0.39
no Manipulation ($\neg M$)	0.26	0.35	0.61
Total	0.65	0.35	1

A higher proportion of images are watermarked, as manipulated images consists only of 60% of the watermarked images. Since the watermark is required to detect and localize manipulations it is implied that only watermarked images undergo malicious manipulations. A sufficient number of watermarked and manipulated images are needed to train the detection and authentication heads in the Forensic Module.

To embed the watermark, the output from the Encoder is multiplied by a scaling factor (wm_scale $\in [0, 1]$) and added to the original image. This scale provides the ability to balance watermark visibility against robustness and forensic performance. The wm_scale parameter can be fixed during training and adjusted as needed during testing.

Two models are trained one with wm_scale = 0.10 and 0.05. Both the models are trained for 10 epochs using the AdamW optimizer [25] (lr $= 2 \times 10^{-4}$, $\beta = (0.9, 0.999)$, weight decay $= 10^{-4}$) with a batch size of 4 and gradient clipping capped at 1.0. The learning rate uses a 2-epoch warmup followed by cosine annealing decaying [26] to 10^{-6}. Each model is trained using Nvidia A100 GPUs. Testing and performance evaluation is performed on Nvidia L4 GPUs. Figure 4 shows the loss vs epoch convergence graph for model trained with wm_scale = 0.10. Time consumed for training and testing is tabulated in Table 4.

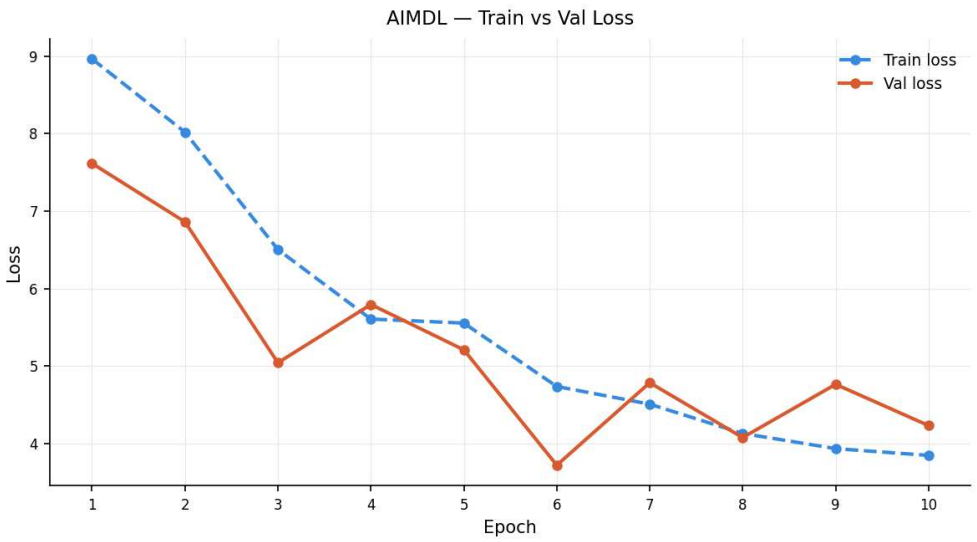


Fig. 4 Epoch vs Training and Validation Loss graph for wm_scale = 0.10 model.

Table 4 Device and time consumption details for training and testings

Task	Device	Time
Training + Validation	A100	102.2 sec / epoch [#Samples: Train: 2100, Val: 300] [Batch size: 4]
Testing	L4	6.66 samples / second 154 millisecond / sample

5 Results and Discussions

For both models, performance metrics are computed by changing wm_scale at test time to different values. Performance metrics for both the models are tabulated in Tables 5 and 6. Figure 5 plots and compares different metrics for both the models at different wm_scales. It can be noted that in the following results, wm_scale of 0 indicates no watermark added. Low performance at wm_scale = 0 indicates that the Forensic Module depends on the watermark for detection and localization.

Table 5 Performance metrics for different watermark scales (changed during testing after training on WM scale of 0.1)

WM_scale	Transparency		Authentication			Detection			Detection (Watermark only)			Localization		
	PSNR	SSIM	Acc	Prec	Rec	Acc	Prec	Rec	Acc	Prec	Rec	IoU	F1	Pixel Acc
0.00	inf	1	0.355	1	0.0026	0.605	1	0.0042	0.3892	1	0.0042	0.0784	0.0983	0.9461
0.01	51.5717	0.9992	0.365	1	0.018	0.615	1	0.0294	0.4046	1	0.0294	0.1418	0.1707	0.9497
0.03	46.9782	0.9985	0.5083	1	0.2397	0.7017	0.9836	0.2521	0.5387	0.9836	0.2521	0.4273	0.4774	0.9661
0.05	42.4001	0.9962	0.785	1	0.6675	0.875	1	0.6849	0.8067	1	0.6849	0.7298	0.7712	0.9838
0.10	36.7034	0.989	0.97	1	0.9536	0.9683	1	0.9202	0.951	1	0.9202	0.88	0.9123	0.9927
0.20	30.7007	0.9695	0.9967	1	0.9948	0.98	1	0.9496	0.9691	1	0.9496	0.909	0.9371	0.994

Table 6 Performance metrics for different watermark scales (changed during testing after training on WM scale of 0.05)

WM_scale	Transparency		Authentication			Detection			Detection (Watermark only)			Localization		
	PSNR	SSIM	Acc	Prec	Rec	Acc	Prec	Rec	Acc	Prec	Rec	IoU	F1	Pixel Acc
0.00	inf	1	0.355	1	0.0026	0.6067	1	0.0084	0.3918	1	0.0084	0.1257	0.1462	0.9483
0.01	51.7544	0.9988	0.3817	1	0.0438	0.64	1	0.0924	0.4433	1	0.0924	0.2979	0.3333	0.9568
0.03	44.8289	0.9938	0.82	1	0.7216	0.895	1	0.7353	0.8376	1	0.7353	0.7782	0.8153	0.984
0.05	41.3545	0.9881	0.9317	1	0.8943	0.9483	0.9952	0.8739	0.9201	0.9952	0.8739	0.8476	0.8831	0.989
0.10	35.5505	0.9699	0.9817	1	0.9716	0.965	1	0.9118	0.9459	1	0.9118	0.8802	0.9116	0.9922
0.20	29.6478	0.9336	0.9933	1	0.9897	0.9633	1	0.9076	0.9433	1	0.9076	0.8647	0.8996	0.9923

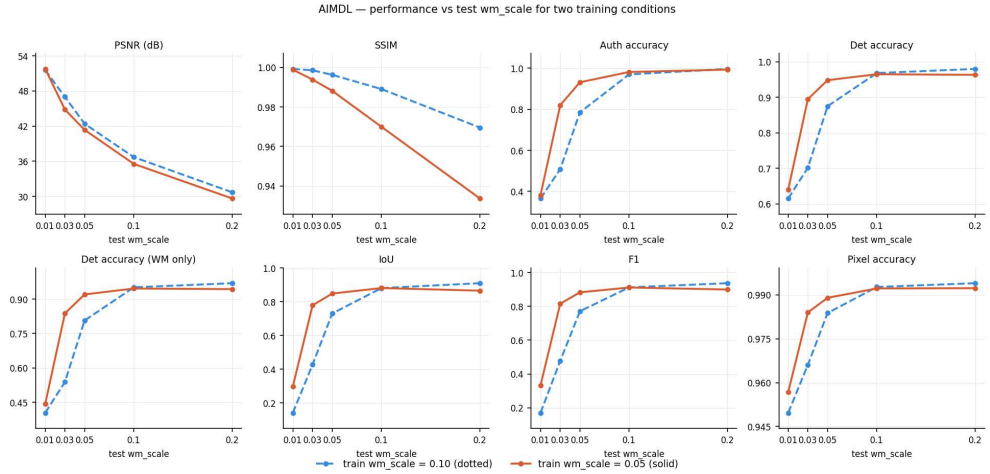


Fig. 5 Comparison of both models (trained with wm_scale = 0.1 and 0.05) at different test time wm_scales.

The figure 5 illustrates a clear trade-off: as the test wm_scale increases, forensic performance (accuracy, IoU, F1) improves, but watermark invisibility (PSNR, SSIM)

consistently degrades. It can also be observed that the model trained with wm_scale = 0.05 reaches maximum forensic performance at lower test wm_scales. There is no significant difference between the two models in case of change in invisibility (PSNR, SSIM) with change in test wm_scales. Confusion matrices for authentication and detection for both the models are plotted in Figures 6 and 7. Figures 8 and 9 showcase sample outputs from both the models. Note that the watermark shown in these sample outputs are before being scaled down to watermark scale.

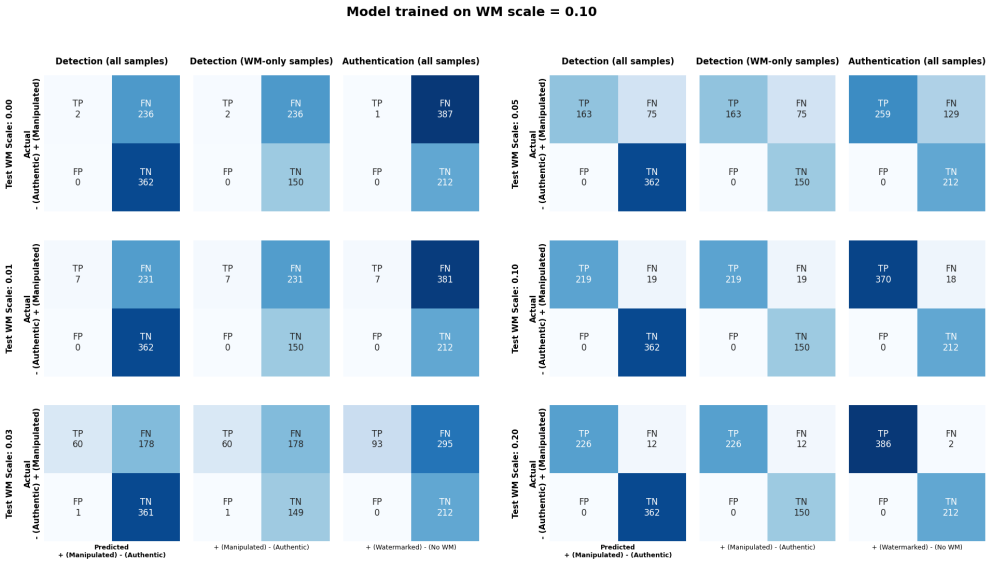


Fig. 6 Confusion matrices for model trained on wm_scale=0.10 for authentication and detection computed for different values of test wm_scales.

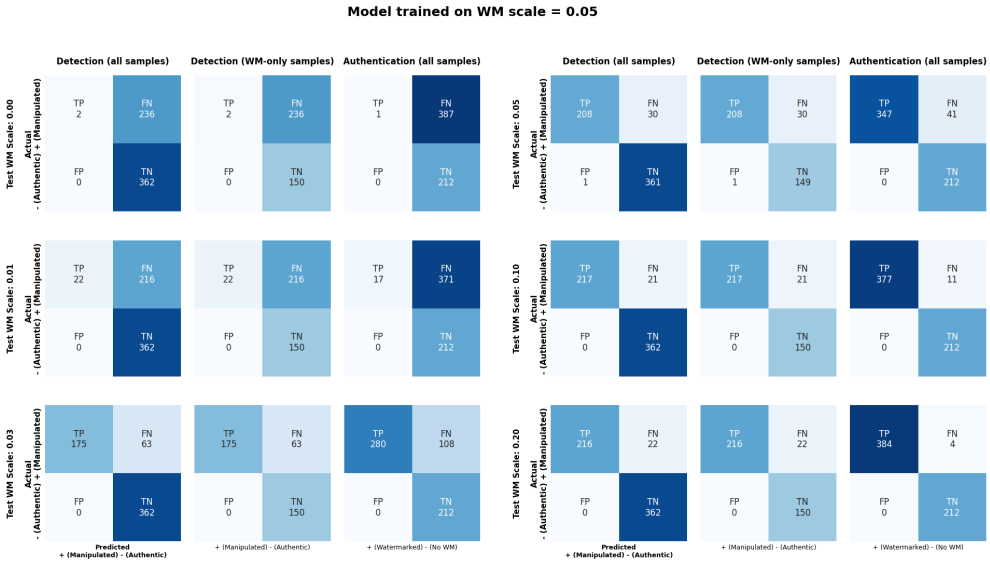


Fig. 7 Confusion matrices for model trained on wm_scale=0.05 for authentication and detection computed for different values of test wm_scales.

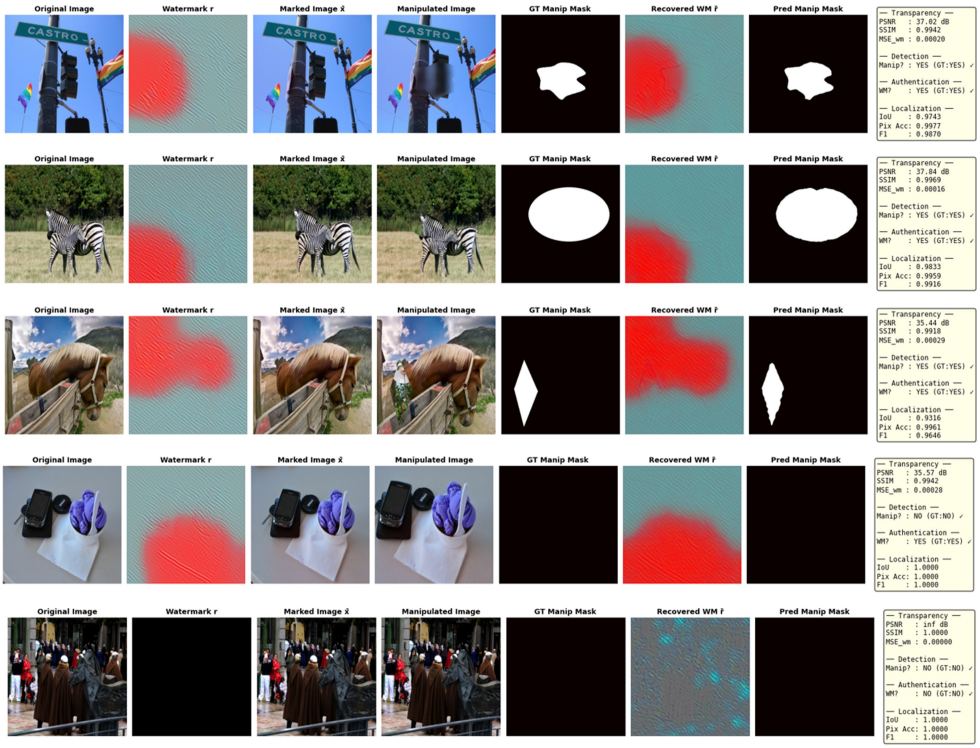


Fig. 8 Sample outputs from model trained on wm_scale = 0.10. (GT = Ground Truth). r represents the watermark before being scaled down by wm_scale

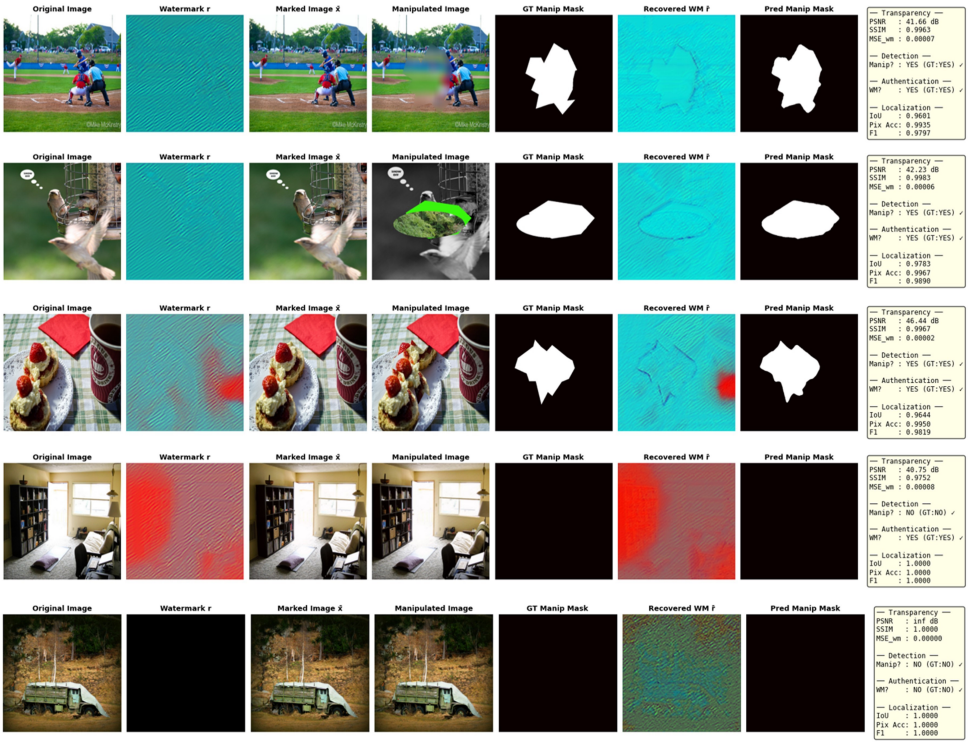


Fig. 9 Sample outputs from model trained on wm_scale = 0.05. (GT = Ground Truth)

6 Conclusions

This study presents a comprehensive and adaptable Active Image Manipulation Detection and Localization (AIMDL) framework leveraging TransUNet architectures. By embedding fragile watermarks at the time of image creation, the proposed system successfully authenticates digital media and provides precise, pixel-level localization of malicious edits such as splicing and copy-move attacks. The integration of specialized Benign Noise and Manipulation modules during training, guided by a diverse set of structural and forensic loss functions, ensures the neural watermark remains resilient to standard processing (like JPEG compression) while instantly fracturing under malicious tampering. Extensive testing across different watermark scaling factors (wm_scale of 0.10 and 0.05) clearly illustrates the fundamental trade-off between visual transparency (measured via PSNR and SSIM) and forensic performance (measured via Accuracy, IoU, and F1-Scores). Notably, the model trained with a lower scaling factor of 0.05 was able to reach maximum forensic efficacy at lower test scales, offering a highly optimized balance of security and imperceptibility.

Declarations

Conflict of Interest

The authors declare that they have no conflict of interest.

Code and Data Availability

Source Code and Data produced will be provided on request. The following publicly datasets are used in this study:

- MSCOCO [21]: available online on request at: https://cocodataset.org
- OpenImageV7 [22]: available online on request at: https://docs.ultralytics.com/datasets/detect/open-images-v7

Author Contribution

All authors contributed equally to the conceptualization, implementation, evaluation, and documentation of this work.

Consent to Publish

All authors consent to the publication of this manuscript.

Funding

This research did not receive any grants/funding.

Ethics Approval

Not Applicable

References

[1] McAndrew, W.P., Speaker, P.J.: Interpol review of forensic science management, 2022–2025. Forensic Sci. Int. Synergy **13**, 100695 (2026) https://doi.org/10.1016/j.fsisyn.2026.100695

[2] Duszejko, P., Walczyna, T., Piotrowski, Z.: Detection of manipulations in digital images: A review of passive and active methods utilizing deep learning. Appl. Sci. **15**(2), 881 (2025) https://doi.org/10.3390/app15020881

[3] Dugelay, J.-L., Rey, C.: A survey of watermarking algorithms for image authentication. EURASIP J. Adv. Signal Process. **2002** (2002) https://doi.org/10.1155/S1110865702204047

[4] Zanardelli, M., Guerrini, F., Leonardi, R., *et al.*: Image forgery detection: a survey of recent deep-learning approaches. Multimed. Tools Appl. **82**, 17521–17566 (2023) https://doi.org/10.1007/s11042-022-13797-w

[5] Chen, J., *et al.*: TransUNet: Rethinking the U-net architecture design for medical image segmentation through the lens of transformers. Med. Image Anal. **97**, 103280 (2024) https://doi.org/10.1016/j.media.2024.103280

[6] Sahu, A.K.: A logistic map based blind and fragile watermarking for tamper detection and localization in images. J. Ambient Intell. Humaniz. Comput. **13**(8), 3869–3881 (2022) https://doi.org/10.1007/s12652-021-03365-9

[7] Abdelhakim, A., Saleh, H.I., Abdelhakim, M.: Fragile watermarking for image tamper detection and localization with effective recovery capability using K-means clustering. Multimed. Tools Appl. **78**, 32523–32563 (2019) https://doi.org/10.1007/s11042-019-07986-3

[8] Mohan, B., Swamy, K.V.: On the use of Schur decomposition for copyright protection of digital images. Int. J. Comput. Electr. Eng. **2**, 781–787 (2010) https://doi.org/10.7763/IJCEE.2010.V2.228

[9] Chen, C.-L., Yao, H.-L., Jian, B.-L.: Discrete wavelet transform sampling for image super resolution. Appl. Artif. Intell. **39** (2025) https://doi.org/10.1080/08839514.2024.2449296

[10] Ikotun, A.M., Ezugwu, A.E., Abualigah, L., Abuhaija, B., *et al.*: K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data. Inf. Sci. **621**, 1–32 (2023) https://doi.org/10.1016/j.ins.2022.11.139

[11] Lefèvre, P., Carré, P., Fontaine, C., *et al.*: Efficient image tampering localization using semi-fragile watermarking and error control codes. Signal Process. **190**, 108342 (2022) https://doi.org/10.1016/j.sigpro.2021.108342

[12] Bouarroudj, R., Souami, F., Bellala, F.Z., Zerrouki, N., Harrou, F., Sun, Y.: Secure and reversible fragile watermarking for accurate authentication and tamper localization in medical images. Comput. Electr. Eng. **123**, 110072 (2025) https://doi.org/10.1016/j.compeleceng.2025.110072

[13] Asnani, V., Yin, N., Hassner, T., Liu, X.: Proactive image manipulation detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 13264–13273 (2022)

[14] Bartolucci, F., Masi, I., Lisanti, G.: Perturb, attend, detect, and localize (PADL): Robust proactive image defense. IEEE Access **13**, 81755–81768 (2025) https://doi.org/10.1109/ACCESS.2025.3565824

[15] Zhao, Y., Liu, B., Zhu, T., Ding, M., Yu, X., Zhou, W.: Proactive image manipulation detection via deep semi-fragile watermark. Neurocomputing **585**, 127593 (2024) https://doi.org/10.1016/j.neucom.2024.127593

[16] Abrar, I., Sheikh, J.A.: DeepMark: A proactive deep learning-based watermarking model for tamper detection and localization across images and videos. Inf. Process. Manage. **63**(4), 104600 (2026) https://doi.org/10.1016/j.ipm.2025.104600

[17] Duszejko, P., Walczyna, T., Piotrowski, Z.: 4KSecure: A universal method for active manipulation detection in images of any resolution. Appl. Sci. **15**(8), 4469 (2025) https://doi.org/10.3390/app15084469

[18] Wang, T., Cheng, H., Liu, M.-H., Kankanhalli, M.: FractalForensics: Proactive deepfake detection and localization via fractal watermarks. In: Proceedings of the 33rd ACM International Conference on Multimedia (2025). https://doi.org/10.1145/3746027.3754544

[19] Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28

[20] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems, vol. 30, pp. 5998–6008 (2017)

[21] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common objects in context. In: European Conference on Computer Vision (ECCV), pp. 740–755 (2014)

[22] Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Malloci, M., Kolesnikov, A., Duerig, T., Ferrari, V.: The open images dataset V4: Unified image classification, object detection, and visual relationship detection at scale. Int. J. Comput. Vis. **128**(7), 1956–1981 (2020)

[23] Voxel51: FiftyOne Dataset Zoo. https://voxel51.com/docs/fiftyone/user_guide/dataset_zoo/index.html (2021)

[24] Balakrishnan, G., Zhao, A., Sabuncu, M.R., Guttag, J., Dalca, A.V.: Voxelmorph: A learning framework for deformable medical image registration. IEEE Transactions on Medical Imaging **38**(8), 1788–1800 (2019) https://doi.org/10.1109/TMI.2019.2897538

[25] Loshchilov, I., Hutter, F.: Decoupled Weight Decay Regularization (2019). https://doi.org/https://arxiv.org/abs/1711.05101

[26] Loshchilov, I., Hutter, F.: Sgdr: Stochastic gradient descent with warm restarts. (2016). https://doi.org/10.48550/arXiv.1608.03983