JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

# DYNAMIC RESIDUAL FORGERY-AWARE DENOISING AND GRADIENT-AWARE PATCH SALIENCY FOR ROBUST IMAGE FORGERY DETECTION

**[1]P.Muniasamy, Dr.B.Srinivasan**

[1]Assistant Professor, Department of Information Technology, Sri Ramakrishna Mission Vidyalaya College of Arts and Science (Autonomous and Affiliated to Bharathiar University), Coimbatore, Tamil Nadu, India.
[2]Associate Professor, Department of Computer Science, Gobi Arts & Science College (Autonomous and Affiliated to Bharathiar University), Gobichettipalayam, Erode, Tamil Nadu, India.

**Abstract:** There has been an increase in the use of digital image tampering due to the availability of advanced image editing software, which has become a significant challenge for the authentication of images and digital forensics. Preprocessing and effective segmentation procedures are required to detect the tampered regions as well as store useful forensic information. This research proposes a robust image forgery detection architecture integrating of the Dynamic Residual Forgery-aware Noise Suppression (DRFNS) for denoising and Gradient-Aware Patch Saliency (GAPS) for segmentation. The DRFNS algorithm performs adaptive patch-based denoising by calculating residual signals and other indicators of forgery like edge density and structural inconsistencies. The method allows high noise suppression in normal areas and preserves traces of manipulation, which are subtle in suspicious areas. Gradient variations, texture anomalies, and color anomalies are then examined and used to compute the patch-level saliency scores on the denoised image by the GAPS segmentation technique. Additionally, K-Means clustering is also included to improve the accuracy of segmentation by having the ability of dividing suspicious regions and normal areas effectively. As shown by experiments, the proposed framework has obtained the highest denoising results by achieving a PSNR of 40.48, SSIM of 0.96, and RMSE of 0.0093. In the case of segmentation, the approach achieved an accuracy of 97.00% and outperformed existing approaches. These findings prove the effectiveness of the proposed framework for reliable digital image forgery detection.
**Index Terms:** Digital Image Forensics, Gradient-Aware Patch Saliency, Image Tampering, Patch-Level Tampering Localization, Residual Signal Reconstruction.

## 1. INTRODUCTION

The authenticity of digital images is vital in the social media, journalism, surveillance, and legal investigations. Nevertheless, the proliferation of advanced image editing software has facilitated the fact that it is now simple to experiment with digital images without any visible evidence. Image splicing and tampering are some of the most difficult forms of manipulation as they add some minor inconsistency to texture, noise definitions, and structural data. Such manipulations can only be detected using the advanced image forensic methods that can detect the smallest artifact that can be used to prove that the image has been forged [1], [2].

The currently used image forgery detection algorithms typically are based on the deep learning designs and statistical feature selection as well as frequency-domain analysis. A number of works use the idea of using demosaicing artifacts and noise discrepancies as the means of identifying the manipulation of images [3]. Convolutional Neural Networks (CNN) and hybrid feature extraction models based on Deep Learning (DL) have also become popular in enhancing the detection accuracy [4-6]. Also, multi-scale edge analysis and self-supervised training mechanisms have been suggested to make the detection of structural inconsistencies in manipulated images more effective [7].

The latest studies have also covered high-order architectures such as attention-based networks, transfer learning systems, and temporal convolutional networks to enhance resilience in forgery detection applications [8], [9]. Noise-guided strategies of learning and residual analysis methods have also been considered, and it was revealed that they are effective in the process of uncovering the traces of manipulation hidden in the structure of images. Additionally, the forged region localization has been performed through machine learning techniques and saliency-based techniques by analyzing the features and patches [10].

Even though this enhancement has been done, there are several limitations to different methods that have been implemented. The smoothing of the process of denoising may also remove minute forensic evidence that may be of critical importance in tracing the areas of interference. Alternatively, also deep neural network models are frequently costly in terms of computer resource, and hence they are not an ideal option in terms of massive analysis. Also, some forms of the segmentation techniques fail to show fine gradient variations and texture discontinuities, and so, they are not good in the localization of complex images [11-13].

To address these issues, this research proposes a two-stage image forgery detector based on the combination of Dynamic Residual Forgery-aware Noise Suppression (DRFNS) denoising and Gradient-Aware Patch Saliency (GAPS) segmentation. DRFNS algorithm can remove any unwanted noise and compression artifact and leaves behind residual signals which may indicate the presence of a manipulation. After the image has been denoised, GAPS segmentation approach to examine gradient variations, edges and patch-based saliency is done with the aim of determining the areas of the image that are suspicious.

This research is motivated by a need to come up with a lean and mean framework that will offer the balance between noise reduction and forensic information retention. The proposed method will use the existing residual-based denoising and gradient-conscious saliency analysis to enhance the accuracy of localization of tampered areas at the same time as being computationally efficient.

The contributions of this work are summarized as follows: (1) introduction of a DRFNS-based denoising mechanism that does not remove important residual forensic data, (2) the creation of a GAPS-based segmentation strategy that allows achieving the accurate localization of manipulated areas, and (3) the incorporation of lightweight feature learning to improve the overall quality of detection.

The rest of this paper is structured in the following way. Section 2 conducts a review of the associated literature on image forgery detection methods. Section 3 uses the presented GAPS segmentation and proposed DRFNS denoising methodology. Section 4 presents the experimental setup and the results of performance evaluation. Lastly, section 5 will conclude the paper and state potential future research directions.

## 2. LITERATURE REVIEW

The background review has explored recent research on deep learning-based techniques of detecting various forms of digital forgery in images, documents as well as biometrical content. These studies were aimed at improving the precision of detection with the help of advanced neural network architectures, optimization techniques along with image enhancement techniques to reveal manipulation artifacts.

Gowda and Pawar (2023) [14] introduced an artificial intelligence architecture of forged regions detection and localization in the video sequence through the application of deep learning. It used the CNNs as the approach to identify the spatial and time inconsistencies in the process of manipulation. The model produced frame-level characteristics and applied techniques of classification to identify tampered frames. The results of the experiment showed that the accuracy of the detection as compared to the traditional methods of video forensics was higher. However, the procedure was equally expensive in computing power in case of long video flows.

Zhao et al. (2021) [15] proposed an image document forgery attack and detection model based on deep learning. In their research, they examined weaknesses of document authentication systems through creating realistic forged ones with processes of neural networks. The model was trained on the texture and structural patterns of documents to identify artifacts of manipulation. The experimental data demonstrates that the given method was useful in detecting forged document areas. However, the framework exhibited weakness in the case of document images that are highly compressed.

Poddar et al. (2020) [16] created a DL method of recognizing offline signatures and detecting forgery. The researchers used convolutional neural networks in order to learn discriminative features of handwritten signatures. The model identified authentic and fake signatures with respect to structural and stroke features. Experimental analyses demonstrated great recognition accuracy on standard signature data sets. Nevertheless, the approach needed huge training datasets in order to be reliable in their performance.

Maashi et al. [17] suggested a hybrid architecture that combines the Reptile Search Algorithm with DL to detect forgery in copy-move images. The feature selection and detection performance were optimized with the help of the optimization algorithm. An image-pattern duplicated through forgery was detected using a deep neural network that analyzed image patterns. The results of the experiment proved to be more accurate and robust than the ordinary machine learning techniques. The method was more complex in computation in optimization, yet it was effective.

Bae et al. (2025) [18] proposed a log-transform histogram equalization method could be used to enhance document forgery detection with deep learning tools. The technique increased the contrast and definition of latent artifacts of tampering prior to feature extraction. The Deep Neural Network (DNN) was then used to categorize genuine and forged documents. Evaluations through experiments showed that there was a greater accuracy of the detection following the application of the suggested method of improvement. Nevertheless, the method was prone to extreme changes in lighting of document images.

Another proposal called ForensicNet was proposed by Tyagi and Yadav (2023) [19], which is a convolutional neural net architecture specifically designed to detect image forgery. Network derived hierarchical spatial characteristics that were used to detect inconsistencies added during tampering. The model was found to be suitable to identify different forms of forgery, such as splicing and copy-move manipulations. It was observed that experimental outcomes had better performance than the current CNN-based forensic approaches. Nonetheless, the model needed a large amount of training data to extrapolate to other sets of images.

In an attempt to enhance the reliability of biometric authentication, Kim (2025) [20] came up with a DL-based facial forgery detection system. The model proposed examined the anomalies of facial texture and artifacts of face-swapping methods. Training a DNN model was trained to differentiate between authentic and fake facial images. The experimental findings indicated that biometric verification under experimental conditions had high detection accuracy. However, the technique had poorer results in working with low-resolution human face images.

Elaskily et al. (2020) [21] suggested a DL architecture of copy-move image forgery detection. This paper utilized CNNs to automatically acquire discriminative features of images without requiring engineering any features manually. The framework identified repeated regions through spatial correlation and similarity pattern. It was found that the proposed model performed better in detecting on standard benchmark datasets. Nevertheless, the performance of the method was poor in case of the highly compressed images.

**Table 1: Summary Table Comparison of State of the art Methods**

| Author & Year | Method / Model Used | Key Concept | Advantages | Limitations |
|---|---|---|---|---|
| Asghar et al. (2022) [22] | Noise and Edge Weighted Local Texture Features | Utilized noise patterns and edge-based local texture descriptors to detect forged regions in images. | Improved detection of subtle tampering artifacts and texture inconsistencies. | Performance decreased when images had heavy compression or smoothing. |
| Shan et al. (2024) [23] | RIFD-Net (Robust Image Forgery Detection Network) | Proposed deep neural network architecture for detecting and localizing image forgery using robust feature extraction. | Achieved high accuracy and robustness across different manipulation types. | Required large datasets and computational resources for training. |
| Nasir et al. (2026) [24] | DnCNN + Multimodal Feature Fusion | Combined DnCNN denoising with multimodal feature fusion to enhance image forgery detection in IoT sensor systems. | Improved detection accuracy by removing noise and combining multiple feature representations. | Increased processing complexity due to multimodal fusion mechanisms. |
| Chakraborty et al. (2024) [25] | Deep Learning + Error Level Analysis | Used error level analysis and noise residual features with deep learning for detecting tampered image regions. | Effective in identifying compression-based manipulation artifacts. | Performance depended on image compression characteristics. |
| Li et al. (2022) [26] | Super-BPD Segmentation + DCNN | Applied superpixel segmentation followed by deep CNN to detect and localize copy-move forgery. | Provided accurate localization of duplicated regions. | Computational cost increased due to segmentation processing. |
| Choudhary et al. (2024) [27] | VGG16 + UNet Model | Integrated VGG16 feature extractor with UNet architecture for forgery detection and segmentation. | Enabled precise pixel-level localization of forged areas. | Model complexity increased training time and resource usage. |
| Al-Nabhani et al. (2026) [28] | REFORGE Ensemble Model | Proposed an ensemble framework combining multiple deep learning models for forgery detection in social network images. | Improved robustness and detection performance through ensemble learning. | Ensemble models increased computational overhead and system complexity. |
| Qazi et al. (2022) [29] | DL-Based Forgery Detection System | Developed a DNN framework to automatically detect manipulated digital images. | Achieved strong detection accuracy without manual feature engineering. | Generalization performance depended on training dataset diversity. |

The table 1 presents the current researches on image forgery detection, including the methods, concepts, benefits and shortcomings of the different DL and feature-based techniques. These works showed that there were remarkable improvements on the detection and localization of manipulated regions, despite the still persistent issues of computational complexity and reliance on large datasets.

## 3. PROPOSED METHODOLOGIES

In this research, emphasize on image fidelity restoration and the ability to correctly localize tampered areas in the images. Denoising is performed on a DRFNS algorithm to reduce noise and retain structural content. The forgery mask is then segmented in a GAPS-based framework with a K-means clustering to get precise forgery mask. The two-stage pipeline is effective based on metrics and visualizations to achieve results in training and testing.

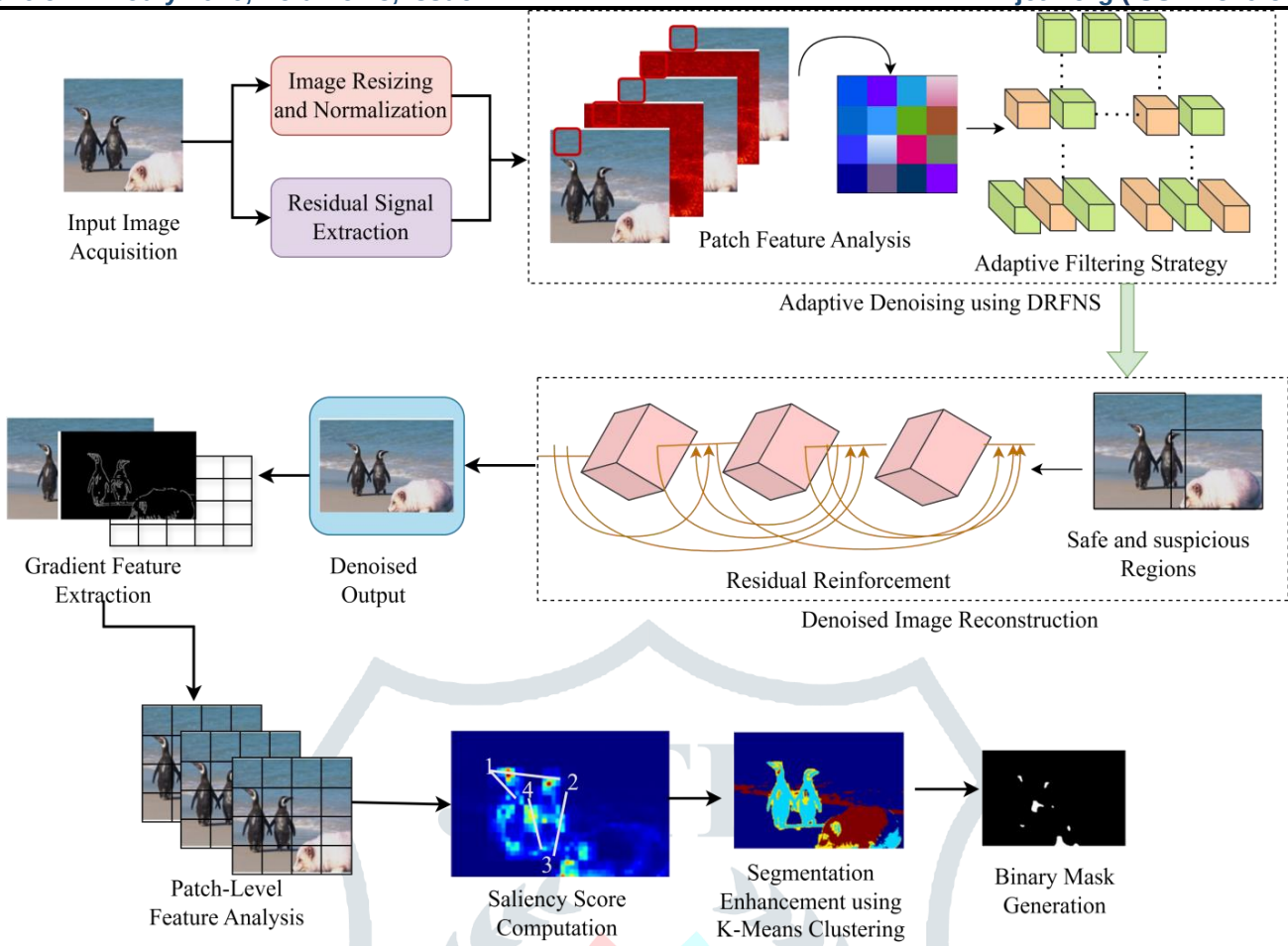


**Figure 1: Two-Stage Residual-Saliency Image Forgery Detection Architecture**

The architecture of Figure 1 includes two steps: DRFNS-based denoising and GAPS-based segmentation. First, DRFNS eliminates noise and compression artifacts, but does not destroy significant forensic residual data in the image. Denoised image is then subjected to GAPS to study the gradient and the texture discontinuity, so that the tampered areas can be localized with greater precision by segmentation.

### 3.1 Dataset Collection

CASIA2 Forgery Detection dataset https://www.kaggle.com/datasets/vortecccc/casia2-forgery-detection is a benchmark dataset of genuine and tampered images which has been widely used in digital image forensics research. It has more than 17,000 photos that demonstrate the manipulations splicing and copy-move forgery. The dataset is structured to be easily explored and is used to train and evaluate machine learning models used to detect image forgery and is under the MIT license.

### 3.2 Denoising using Dynamic Residual Forgery-aware Noise Suppression (DRFNS)

The denoising process is aimed at removing artifacts of noise and compression to original and potentially altered images and preserving small forensic hints that could be used to show the manipulation. To this end, an algorithm like DRFNS algorithm is used as a preprocessing step. First, residual signals are obtained by differencing the image with a blurred version of the original image and that in turn reveals high-frequency contents like edges, change of texture, and potential tampering artifacts. These residual signals contain useful forensic information that could otherwise be removed during aggressive denoising. The image is then subdivided into homogeneous, non-overlapping patches so that local analysis is performed instead of using a homogeneous filter to analyze the entire image. The patches are checked against the forgery signs of irregular edge density, the block inconsistencies, or the irregular texture sources. On the basis of this analysis, a forgery score is allocated to each patch to indicate that forgery content is likely to be found in the patch. Patches that are low-scoring in forgery are safe regions and they are given a more robust Gaussian smoothing to effectively eliminate noise and compression artifacts. On the other hand, suspicious patches are subjected to weaker filtering to avoid the elimination of valuable evidence of tampering. Moreover, the left-over signals of these suspicious patches are also partially introduced back with the help of a weighted combination technique. This action will be taken to make sure that any crucial structural evidence and traces of manipulations are not hidden to prevent additional forensic examination. DRFNS balances between the important tasks of reducing noise effectively and maintaining evidence of manipulation by adaptively controlling the strength of the denoising process, depending on the likelihood of patch-level forgery. The resulting denoised image preserves the necessary forensic information and is used as input by the following stage of segmentation, in which possible areas of tampering are identified.

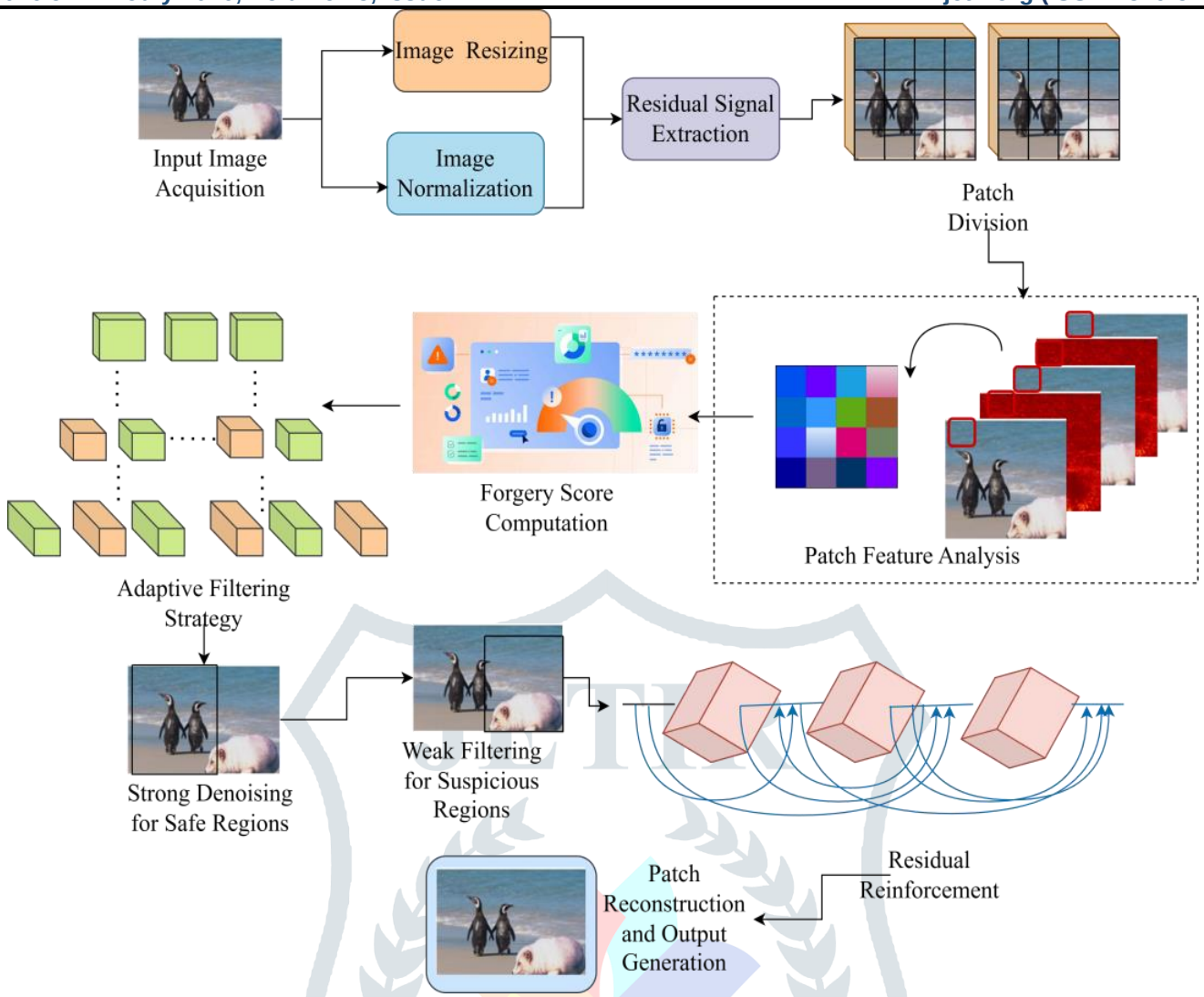


**Figure 2: Denoising Architecture using DRFNS**

In figure 2 illustrates the DRFNS denoising architecture, the first approach extracts residual signals of input image, which shows the noise patterns and possible traces of manipulation. This is further followed by division of the image into patches an adaptive filtering on forgery probability is used to remove noise but retain suspicious structural information. The reconstructed patches are then the processed patches that are finally reconstructed to produce an image that has been denoised but has not lost any essential forensic data to allow further segmentation. For the input image $I$:

$$R = I - B_\sigma(I) \qquad (1)$$

Where, $R$ is a residual image capturing high-frequency components such as edges and tampering artifacts, $I$ is original input image (RGB or grayscale), $B_\sigma(I)$ is gaussian-blurred version of $I$with standard deviation $\sigma$, $\sigma$ is standard deviation of Gaussian kernel; controls smoothing strength. Equation (1) is used to subtracting the blurred image isolates high-frequency components such as edges, noise, and block artifacts.

$$P_{i,j} = I[y_i : y_i + P, x_j : x_j + P] \qquad (2)$$

Where, $P_{i,j}$ is patch at row $i$and column $j$, $P$ is patch size (number of pixels per side), $y_i, x_j$ are coordinates of the top-left pixel of patch $P_{i,j}$. Equation (2) allows localized analysis for denoising and forgery detection.

$$S_{i,j} = \frac{\sum E_{i,j}}{P^2} \qquad (3)$$

Where, $S_{i,j}$ is Forgery score of patch $P_{i,j}$, $E_{i,j}$ is binary edge map (0 or 1) of the patch, computed using Canny edge detection, $P^2$ is total number of pixels in the patch (normalizes edge count). In equation (3) high edge density in uniform areas may indicate tampering.

$$\text{Region}_{i,j} = \begin{cases} \text{Safe} & S_{i,j} < T \\ \text{Suspicious} & S_{i,j} \geq T \end{cases} \qquad (4)$$

Where, $\text{Region}_{i,j}$ is label assigned to the patch: Safe or Suspicious, $T$ is forgery score threshold; separates normal from tampered patches. In equation (4) safe patches are denoised aggressively; suspicious patches are lightly filtered.

$$\hat{P}_{i,j} = \begin{cases} G_{\sigma_s}(P_{i,j}) & \text{if Safe} \\ G_{\sigma_r}(P_{i,j}) + \alpha R_{i,j} & \text{if Suspicious} \end{cases} \qquad (5)$$

Where, $\hat{P}_{i,j}$ is denoised patch, $G_{\sigma_s}$is gaussian filter with strong smoothing ($\sigma_s$) for safe patches. $G_{\sigma_r}$is gaussian filter with weak smoothing ($\sigma_r$) for suspicious patches. $\alpha$ is weighting factor to combine residual $R_{i,j}$into suspicious patches, $R_{i,j}$ is residual corresponding to patch $P_{i,j}$. Equation (5) is used to protect tampering evidence in suspicious patches.

$$\hat{I} = \bigcup_{i,j} \hat{P}_{i,j} \qquad (6)$$

Where, $\hat{I}$ is reconstructed denoised image, $\bigcup_{i,j} \hat{P}_{i,j}$ is combination of all denoised patches into the full image. Equation (6) ensures the full image is restored from processed patches.

**Algorithm 1: Dynamic Residual Forgery-aware Noise Suppression (DRFNS)**

Input: I is Input image (possibly tampered or compressed), PATCH_SIZE is size of image patches (e.g., 32x32), SAFE_SIGMA is Gaussian sigma for safe regions, SUSP_SIGMA is gaussian sigma for suspicious regions, ALPHA is weight for residual preservation
Step 1: Convert I to a suitable format (grayscale or YCbCr)
Step 2: Compute residual image:
I_blur = GaussianBlur(I, sigma=1.0)
R = |I - I_blur|
Step 3: Divide I into non-overlapping patches of size PATCH_SIZE
Step 4: For each patch P in I:
a. Convert P to grayscale
b. Compute edge map (e.g., using Canny or Sobel)
c. Compute forgery score F = mean(edge_density or block inconsistencies)
d. If F < threshold:
P_filtered = GaussianBlur(P, SAFE_SIGMA)
Else:
P_filtered = GaussianBlur(P, SUSP_SIGMA)
P_filtered = ALPHA * P_filtered + (1-ALPHA) * R_patch
e. Replace P in I with P_filtered
Step 5: Return I_denoise = reconstructed image from all patches
Output: I_denoise is denoised image preserving tampering evidence

Algorithm 1 DFRNS algorithm is the image denoising algorithm that analyses the residues and edge patterns of local patches. It applies intensive filtering on the safe areas and minimizes filtering and residual reinforcement in the suspicious areas so that the possible hints of tampering can be left unfiltered.

### 3.3 Segmentation using Gradient-Aware Patch Saliency Segmentation (GAPS)

Gradient-Aware Patch Saliency (GAPS) is a segmentation method that detects the areas that could have been manipulated by examining the structural and texture differences in the denoised image. The denoised image of the last step is first partitioned into non-overlapping patches, which later allow analyzing specifics of the image locally. Gradient information in the horizontal and vertical directions is calculated in every patch to sum up the differences in edges and structural patterns. The gradients are used to point out areas where there is a sudden shift in intensity and this could be a sign of manipulation limits. The density of edges and irregularities of texture in each patch are then assessed in order to determine the probability of tampering.

Besides gradient-based features, the color difference between adjacent patches is examined, to identify unnatural transitions in case of splicing processes and editing methods on images. Depending on these structural, texture and color characteristics, a combined saliency score is computed on each patch and this value indicates a likelihood that the area has been altered. The calculated saliency scores are scaled to ensure the same scale is used throughout the whole image. In an effort to enhance the process of segmentation, K-Means clustering is performed on the values of saliency and the patches are grouped together in clusters of normal and suspicious areas. This clustering is an improvement of segmentation since the high- and low-saliency patches are automatically separated without necessarily using a fixed threshold. Having done the clustering, the suspicious cluster is translated into a binary mask which points out the possible tampered areas. Simple post-processing eliminates small isolated regions and noise artifacts resulting in less false positives.

Lastly, patch-level masks are all combined to generate a complete segmentation mask that distinctively identifies areas that have been manipulated in the image. This improved segmentation output can give a stable depiction of the suspected tampering areas and serves as the input in the analysis and assessment in the proposed structure.

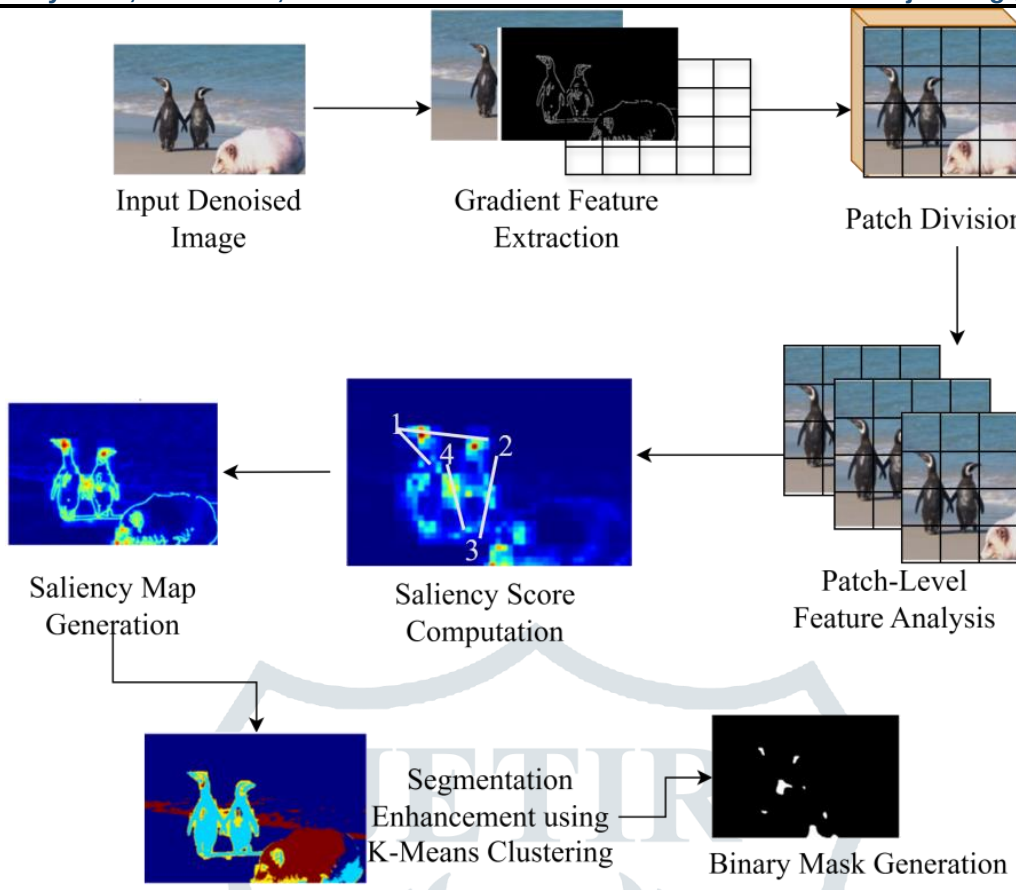


**Figure 3: Segmentation Architecture using GAPS**

Figure 3 GAPS segmentation architecture examines the denoised image by decomposing an image into patches and calculates gradient-based saliency features to identify structural discrepancies. These attributes are applied to approximate tampering probability of each patch and the K-means clustering is used to optimize the segmentation of normal and suspicious regions. Lastly, the patches are grouped together to yield the segmentation mask which indicates the identified forged regions.

$$G_x = \frac{\partial \hat{I}}{\partial x}, G_y = \frac{\partial \hat{I}}{\partial y}, M = \sqrt{G_x^2 + G_y^2} \quad (7)$$

Where, $G_x, G_y$ are Gradients along x and y directions, $M$ is gradient magnitude map, highlights edges and irregular textures. Equation (7) highlights edges and potential tampered boundaries.

$$P_{i,j} = M[y_i : y_i + P, x_j : x_j + P] \quad (8)$$

Where, $P_{i,j}$ is gradient magnitude patch at coordinates $(i, j)$, $P$ is patch size. Equation (8) is used for patch extraction.

$$\mu_{i,j} = \frac{1}{P^2} \sum_{y=0}^{P-1} \sum_{x=0}^{P-1} P_{i,j}(x, y) \quad (9)$$

Where, $\mu_{i,j}$ is mean gradient in patch $P_{i,j}$. Equation (9) quantifies the overall edge strength.

$$\sigma_{i,j} = \sqrt{\frac{1}{P^2} \sum_{y=0}^{P-1} \sum_{x=0}^{P-1} (P_{i,j}(x, y) - \mu_{i,j})^2} \quad (10)$$

Where, $\sigma_{i,j}$ is measures texture variability in the patch. In equation (10) high $\sigma$ often indicates tampered edges.

$$C_{i,j} = \frac{1}{P^2} \sum_{y=0}^{P-1} \sum_{x=0}^{P-1} \| I_{i,j}(x, y) - \bar{I}_{neigh} \| \quad (11)$$

Where, $C_{i,j}$ is color divergence between patch and surrounding patches. $\bar{I}_{neigh}$ is mean RGB color of neighboring patches. Equation (11) detects unnatural color changes from splicing.

$$S_{i,j} = w_1 \cdot \text{mean}(M_{i,j}) + w_2 \cdot \text{std}(M_{i,j}) + w_3 \cdot \text{color\_div}_{i,j} \quad (12)$$

Where, $S_{i,j}$ is saliency score for patch $P_{i,j}$, $M_{i,j}$ is gradient magnitude patch, $\text{mean}(M_{i,j})$ is average gradient magnitude of patch, $\text{std}(M_{i,j})$ is standard deviation of gradient magnitude; measures texture irregularity, $\text{color\_div}_{i,j}$ is color divergence from surrounding patches (detects splicing). $w_1, w_2, w_3$ are weight coefficients controlling importance of each term. Equation (12) combines texture and color irregularities to detect tampered areas.

$$\tilde{S}_{i,j} = \frac{S_{i,j} - \min(S)}{\max(S) - \min(S)} \quad (13)$$

Where, $\tilde{S}_{i,j}$ is normalized saliency (0–1 range), $\min(S), \max(S)$ are minimum and maximum saliency values across all patches. Equation (13) makes saliency comparable across patches for mask generation.

$$Mask_{i,j} = \begin{cases} 1 & \tilde{S}_{i,j} > \tau \\ 0 & \tilde{S}_{i,j} \leq \tau \end{cases} \quad (14)$$

Where, $Mask_{i,j}$ is binary mask for patch $P_{i,j}$(1 = tampered, 0 = normal), $\tau$ is threshold for converting normalized saliency to binary mask. Equation (14) produces a binary mask highlighting likely tampered regions.

$$Mask = \bigcup_{i,j} Mask_{i,j} \quad (15)$$

Where, $Mask$ complete segmentation mask for the input image. Equation (15) combines all patch masks into full resolution.

**Algorithm 2: Gradient-Aware Patch Saliency Segmentation (GAPS)**

Input: I is preprocessed (denoised) image, PATCH_SIZE is size of image patches (e.g., 16x16 or 32x32), SAL_THRESHOLD is saliency threshold for segmentation
Steps:
Step 1: Convert I to grayscale
Step 2: Compute gradient magnitude image G:
    G = sqrt(Sobel_x(I)^2 + Sobel_y(I)^2)
Step 3: Initialize saliency map S_map = zeros_like(I)
Step 4: Divide I into non-overlapping patches of size PATCH_SIZE
Step 5: For each patch P in I:
    a. Compute patch gradient mean: g_mean = mean(G_patch)
    b. Compute texture/edge divergence metrics if needed
    c. Assign saliency score S_patch = g_mean + alpha * texture_metric
    d. Fill corresponding region in S_map with S_patch
Step 6: Normalize S_map to range [0,1]
Step 7: Generate binary mask:
    Mask = (S_map > SAL_THRESHOLD) ? 1 : 0
Step 8: Return Mask and S_map
Output: Mask is binary segmentation mask highlighting suspicious regions, S_map is saliency map of the image

The GAPS algorithm 2 divides an image by calculating the saliency scores of gradients and texture-based saliency of each patch. The saliency patches are labeled as high-saliency indicating that they may have been tampered with and the remaining parts are termed as low-saliency and are ignored, virtually isolating the suspicious parts.

## 4. RESULTS AND DISCUSSIONS

The Results and Discussion section compares the efficiency of the proposed structure as related to denoising and segmentation accuracy. The results of the proposed DRFNS denoising algorithm and GAPS segmentation method are compared with other existing methods to obtain experimental results. Quantitative evaluation is performed using performance metrics of Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) and Root Mean Square Error (RMSE), accuracy, precision, recall and F1-score. The findings reveal the performance of the proposed procedures in the elimination of noise with the maintenance of the forensic information and the ability to recognize the areas of tampering correctly. The results obtained are compared and analyzed in details as discussed in the subsequent subsections.

### 4.1 Performance Evaluation

The effectiveness of the proposed denoising and segmentation framework is evaluated with the help of the conventional quantitative indicators in the performance evaluation. The proposed algorithms are contrasted with some of the existing techniques to show that they are effective. The denoising DRFNS is under comparison with Fast Fourier Transformation - Discriminative Robust Local Binary Patterns (FFTDRLBP) [22], Robust Image Forgery Detection Network (RIFD-Net) [23], Deep Convolutional Neural Network (DnCNN) [24], and Dual branch CNN (DbCNN) [25], the proposed segmentation GAPS algorithm is compared with DCNN [26], Visual Geometry Graph 16-U shaped CNN (VGG16-UNet) [27], Robust Ensample Forgery Detection Algorithm (REFORGE) [28], You Only Look Once-CNN (YOLO-CNN) [29]. The findings underscore the fact that the suggested method can be used to obtain better image reconstruction and precise localization of tampering.

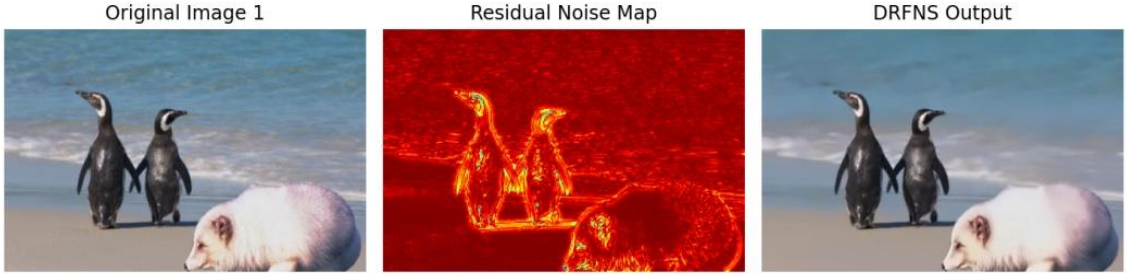


**Figure 4: Denoised Image 1**

Figure 4 shows the process of denoising the input image 1 at DRFNS, the original image is initially used to create a residual noise map that emphasizes the high-frequency features and possible manipulation indicators. Based on this information, DRFNS algorithm removes noise and also conserves key structural information which results in a cleaner image that can be processed further.

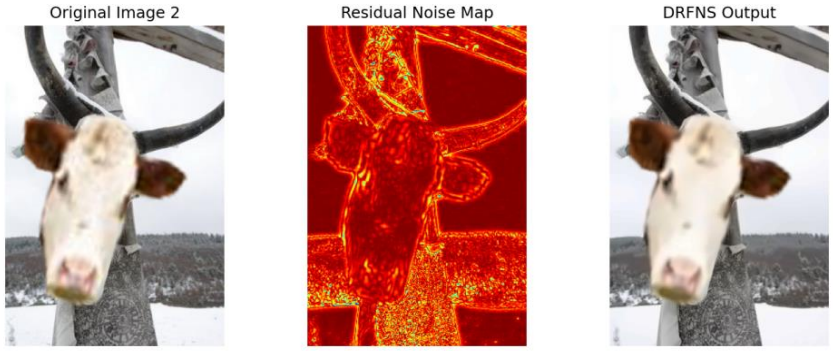


**Figure 5: Denoised Image 2**

Figure 5 shows the process of denoising the input image at DRFNS, the original image is initially used to create a residual noise map that emphasizes the high-frequency features and possible manipulation indicators. Based on this information, DRFNS algorithm removes noise and also conserves key structural information which results in a cleaner image that can be processed further.

**Table 2: Denoised Algorithms Performance Comparison**

| Algorithm | PSNR | SSIM | RMSE |
|---|---|---|---|
| FFTDRLBP | 34.62 | 0.89 | 0.0215 |
| RIFD-Net | 36.78 | 0.91 | 0.0172 |
| DnCNN | 38.12 | 0.93 | 0.0136 |
| DbCNN | 39.05 | 0.94 | 0.0111 |
| DRFNS | 40.480 | 0.96 | 0.0093 |

According to Table 2, the proposed DRFNS denoising algorithm has the highest PSNR and SSIM values and the lowest RMSE values as compared to the current algorithms like FFTDRLBP, RIFD-Net, DnCNN, and DbCNN. This implies that DRFNS does a good job in eliminating noise without affecting the significant structural and forensic information in the image.

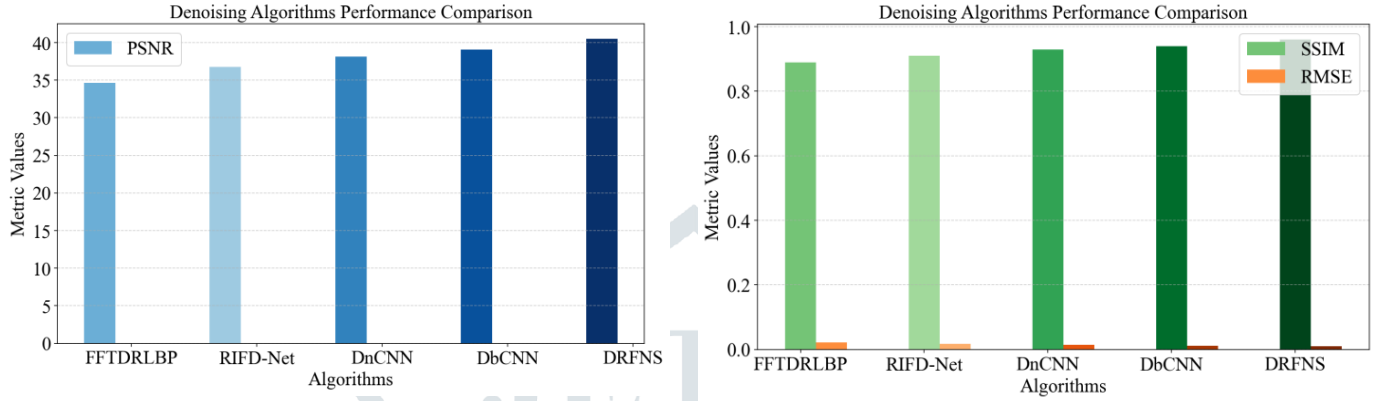


**Figure 6: Denoised Performance Comparison Chart**

Figure 6 shows the denoised algorithms performance comparison using PSNR, SSIM, and RMSE metrics. In this chart the x-axis shows the denoised algorithms and the y-axis show the performance metric values.

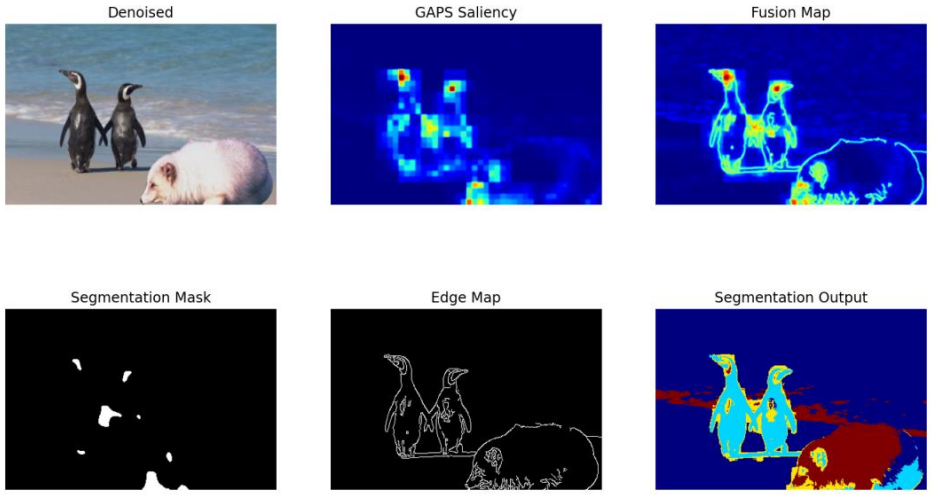


**Figure 7: Segmentation Output Image 1**

Figure 7 illustrates the segmentation process of using GAPS in which the denoised image 1 is processed to produce a saliency map which identifies the regions with high structural and gradient differences. The features are then further refined with edge detection and clustering to give the final segmentation mask with high accuracy on the prominent objects and suspicious regions in the image.

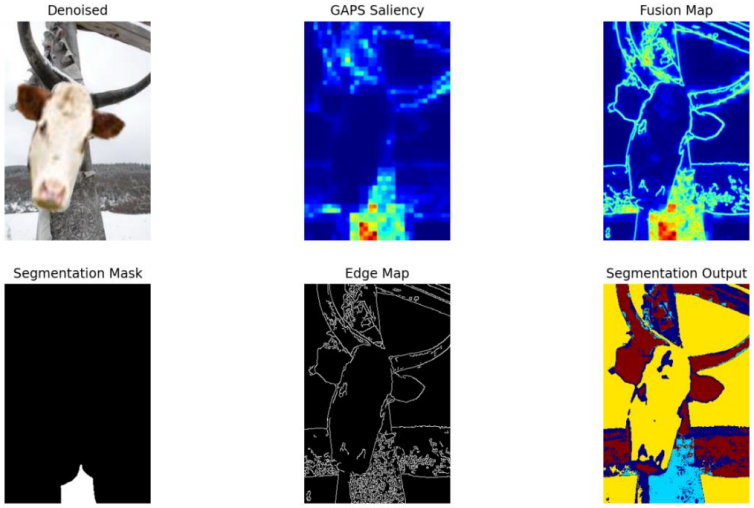


**Figure 8: Segmentation Output Image 2**

Figure 8 illustrates the segmentation process of using GAPS in which the denoised image 2 is processed to produce a saliency map which identifies the regions with high structural and gradient differences. The features are then further refined with edge detection and clustering to give the final segmentation mask with high accuracy on the prominent objects and suspicious regions in the image.

**Table 3: Segmentation Algorithms Performance Comparison**

| Method | Accuracy % | Precision % | Recall % | F1-Score % |
|---|---|---|---|---|
| DCNN | 88.72 | 86.15 | 84.93 | 85.53 |
| VGG16-UNet | 91.48 | 89.67 | 88.12 | 88.88 |
| REFORGE | 93.26 | 92.04 | 90.33 | 91.18 |
| YOLO-CNN | 94.85 | 93.92 | 92.76 | 93.33 |
| GAPS | 97.00 | 96.34 | 94.44 | 95.91 |

Table 3 shows that the proposed GAPS segmentation methodology, it has the greatest level of accuracy and precision compared to other existing methods including DCNN, VGG16-UNet, REFORGE, and YOLO-CNN. This result indicates that gradient-aware patch saliency is effective to detect fake regions with high accuracy and minimize false saliencies.

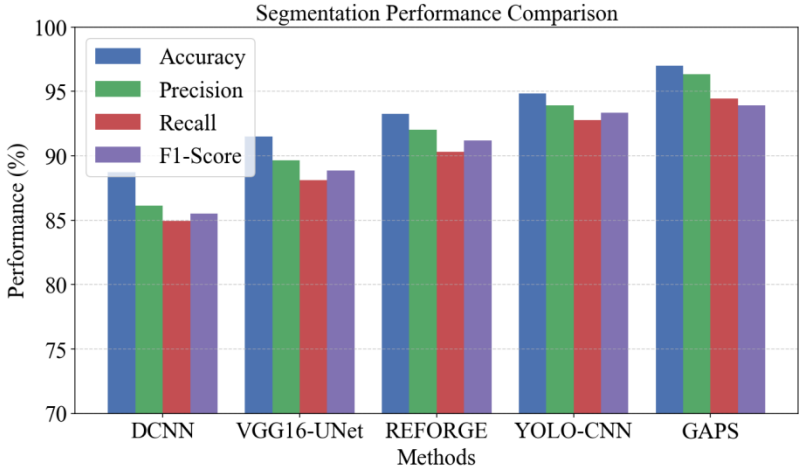


**Figure 9: Segmentation Performance Comparison Chart**

Figure 9 illustrates the performance comparison of segmentation algorithms using segmentation performance metrics such as accuracy, precision, recall, and F1-score. In this chart the x-axis shows the segmentation algorithms and the y-axis shows the performance values.

### 4.2 Discussions

As the experimental analysis demonstrates, the proposed structure of DRFNS-GAPS is not only efficient in regard to image denoising, but also image tampered region segmentation. A mixture of forgery-sensitive denoising and gradient based segmentation process allows the framework to remove noise and compression artifacts and preserves the subtle traces of forensic manipulation. DRFNS does not apply a smooth over the whole image like the traditional denoising techniques which apply it at the local level, a patch. The benefit of this is that it enables potent noise denial in the location with no worthwhile signal, and maintains important residual signal in potentially worthwhile location, which is essential to enable dependable forensic evaluation.

The success of Gradient-Aware Patch Saliency (GAPS) method also is shown by the results of the experiment. The method can identify the boundaries of the altered regions by taking differences in gradient, structural differences and anomalies of the texture at the patch level. The K-Means clustering is brought into the process of segmentation to enhance the accuracy of the segregation between normal and suspicious patches to minimize false detections and localization of tampered areas.

The second advantage of the proposed structure is that the suggested framework is modular wherein the denoising and the segmentation stages complement one another. The quality of images is reflected in the denoising step and the indicators of manipulation are maintained and eventually results in the higher performances of the segmentation process. It is an integration of the two phases that contributes to the general power of the suggested system.

Even though effective, the framework may be problematic in situations when the tampered areas are extremely small or highly compressed and the number of clues left behind is minimal. Additional studies and development in the future can be done to include more complex deep learning models or attention-based models to increase the precision of the segmentation and implement the process to more complex multimedia forgery processes.

## 5. CONCLUSION

This research has presented an effective framework of forgery of digital images through a combination of DRFNS denoising and GAPS partitioning. The solution proposed would be to strip off noise and compression artifacts whilst still preserving critical forensic evidence that would indicate that it has been manipulated. DRFNS is an adaptive patch algorithm, a patch-based filtering algorithm based on the idea that the residual signal and forgery indicators allow control of the intensive denoising of safe areas and low-profile tampering indicators of potential suspicious areas. The experimental results show that DRFNS has PSNR of 40.48, SSIM of 0.96, and RMSE of 0.0093, and this is more impressive compared to the current methods, such as FFTDRLBP, RIFD-Net and DnCNN and DbCNN. At localization of tampered regions, GAPS segmentation algorithm is used to analyze gradient change, texture irregularity and color irregularity changes between patches of the image. The K-Means clustering is also enhanced through the integration that enhances the accuracy in segmentation since the suspicious and the normal areas can be separated sufficiently. The results of the segmentation obtained prove that the proposed approach has the accuracy of 97.00%, the precision of 96.34%, the recall of 94.44%, and the F1-score of 93.91%, which are higher than DCNN, VGG16-UNet, REFORGE, and YOLO-CNN models. These results prove the fact that the mentioned framework is effective to balance between noise removing and the preservation of forensic evidence. Overall, DRFNS-GAPS framework is an effective and potent mechanism in the identification of manipulated regions in digital photographs and has a great likelihood of being adopted in digital image forensics and multimedia security.

**REFERENCES**


[1] Bammey, Q., von Gioi, R. G., & Morel, J. M. (2022). Forgery detection by internal positional learning of demosaicing traces. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 328-338).

[2] Zhu, J., Li, D., Fu, X., Yang, G., Huang, J., Liu, A., & Zha, Z. J. (2024, March). Learning discriminative noise guidance for image forgery detection and localization. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 38, No. 7, pp. 7739-7747).

[3] Gardella, M., Musé, P., Morel, J. M., & Colom, M. (2021). Forgery detection in digital images by multi-scale noise estimation. *Journal of Imaging*, *7*(7), 119.

[4] Mehrjardi, F. Z., & Zarchi, M. S. (2026). A hybrid model for image forgery detection using deep learning with block and keypoint methods. *Scientific Reports*. 476.

[5] Ali, S. S., Ganapathi, I. I., Vu, N. S., Ali, S. D., Saxena, N., & Werghi, N. (2022). Image forgery detection using deep learning by recompressing images. *Electronics*, *11*(3), 403.

[6] Khalil, A. H., Ghalwash, A. Z., Elsayed, H. A. G., Salama, G. I., & Ghalwash, H. A. (2023). Enhancing digital image forgery detection using transfer learning. *IEEE Access*, *11*, 91583-91594.

[7] Zhang, H., Zeng, J., & Yang, J. (2025). Multi-Scale Edge-Guided Image Forgery Detection via Improved Self-Supervision and Self-Adversarial Training. *Electronics*, *14*(9), 1877.

[8] Wang, J., Lei, J., Li, S., & Zhang, J. (2025). STA-3D: Combining Spatiotemporal Attention and 3D Convolutional Networks for Robust Deepfake Detection. *Symmetry*, *17*(7), 1037.

[9] Ghai, A., Kumar, P., & Gupta, S. (2024). A deep-learning-based image forgery detection framework for controlling the spread of misinformation. *Information Technology & People*, *37*(2), 966-997.

[10] Thakur, A., & Jindal, N. (2018, December). Machine learning based saliency algorithm for image forgery classification and localization. In *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)* (pp. 451-456). IEEE.

[11] Anwar, S., & Barnes, N. (2019). Real image denoising with feature attention. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 3155-3164).

[12] Elaskily, M. A., Alkinani, M. H., Sedik, A., & Dessouky, M. M. (2021). Deep learning based algorithm (ConvLSTM) for copy move forgery detection. *Journal of Intelligent & Fuzzy Systems*, *40*(3), 4385-4405.

[13] Dhariwal, P., & Nichol, A. (2021). Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, *34*, 8780-8794.

[14] Gowda, R., & Pawar, D. (2023). Deep learning-based forgery identification and localization in videos. *Signal, Image and Video Processing*, *17*(5), 2185-2192.

[15] Zhao, L., Chen, C., & Huang, J. (2021). Deep learning-based forgery attack on document images. *IEEE Transactions on Image Processing*, *30*, 7964-7979.

[16] Poddar, J., Parikh, V., & Bharti, S. K. (2020). Offline signature recognition and forgery detection using deep learning. *Procedia Computer Science*, *170*, 610-617.

[17] Maashi, M., Alamro, H., Mohsen, H., Negm, N., Mohammed, G. P., Ahmed, N. A., ... & Alsaid, M. I. (2023). Modeling of reptile search algorithm with deep learning approach for copy move image forgery detection. *IEEE Access*, *11*, 87297-87304.

[18] Bae, Y. Y., Cho, D. J., & Jung, K. H. (2025). A new log-transform histogram equalization technique for deep learning-based document forgery detection. *Symmetry*, *17*(3), 395.

[19] Tyagi, S., & Yadav, D. (2023). ForensicNet: Modern convolutional neural network-based image forgery detection network. *Journal of Forensic Sciences*, *68*(2), 461-469.

[20] Kim, H. (2025). Novel Deep Learning-Based Facial Forgery Detection for Effective Biometric Recognition. *Applied Sciences*, *15*(7), 3613.

[21] Elaskily, M. A., Elnemr, H. A., Sedik, A., Dessouky, M. M., El Banby, G. M., Elshakankiry, O. A., ... & Abd El-Samie, F. E. (2020). A novel deep learning framework for copy-moveforgery detection in images. *Multimedia Tools and Applications*, *79*(27), 19167-19192.

[22] Asghar, K., Saddique, M., Hussain, M., Bebis, G., & Habib, Z. (2022). Image forgery detection using noise and edge weighted local texture features. *Advances in Electrical and Computer Engineering*, *22*(1), 57-68.

[23] Shan, W., Zou, D., Wang, P., Yue, J., Liu, A., & Li, J. (2024). RIFD-Net: A robust image forgery detection network. *IEEE Access*, *12*, 20326-20340.

[24] Nasir, N., Waseem, S. S., Bilal, M., & Hassan, S. R. (2026). IoT-Oriented Security for Small Sensor Systems Using DnCNN Denoising and Multimodal Feature Fusion for Image Forgery Detection. *Sensors*, *26*(4), 1172.

[25] Chakraborty, S., Chatterjee, K., & Dey, P. (2024). Detection of image tampering using deep learning, error levels and noise residuals. *Neural Processing Letters*, *56*(2), 112.

[26] Li, Q., Wang, C., Zhou, X., & Qin, Z. (2022). Image copy-move forgery detection and localization based on super-BPD segmentation and DCNN. *Scientific Reports*, *12*(1), 14987.

[27] Choudhary, R. R., Paliwal, S., & Meena, G. (2024). Image forgery detection system using VGG16 UNET model. *Procedia Computer Science*, *235*, 735-744.

[28] Al-Nabhani, Y., Kamsin, A., Nizam, M., & Al Ruqeishi, K. (2026). REFORGE: A Robust Ensemble for Image Forgery Detection and Localization in Social Network Images. *IEEE Access*, *14*, 8773-8788.

[29] Qazi, E. U. H., Zia, T., & Almorjan, A. (2022). Deep learning-based digital image forgery detection system. *Applied Sciences*, *12*(6), 2851.