Journal of Electronic Research and Application, 2025, Volume 9, Issue 5
http://ojs.bbwpublisher.com/index.php/JERA
ISSN Online: 2208-3510
ISSN Print: 2208-3502

An Image Manipulation Localization Method Based on Dual-Branch Hybrid Convolution

Chengliang Yan[1], Lei Zhang[2]*, Minhui Chang[2]

[1]School of Computer Science and Technology, Taiyuan Normal University, Jinzhong 030619, Shanxi, China
[2]School of Mathematics and Information Technology, Yuncheng University, Yuncheng 044000, Shanxi, China

**Author to whom correspondence should be addressed.*

Abstract: In existing image manipulation localization methods, the receptive field of standard convolution is limited, and during feature transfer, it is easy to lose high-frequency information about traces of manipulation. In addition, during feature fusion, the use of fixed sampling kernels makes it difficult to focus on local changes in features, leading to limited localization accuracy. This paper proposes an image manipulation localization method based on dual-branch hybrid convolution. First, a dual-branch hybrid convolution module is designed to expand the receptive field of the model to enhance the feature extraction ability of contextual semantic information, while also enabling the model to focus more on the high-frequency detail features of manipulation traces while localizing the manipulated area. Second, a multi-scale content-aware feature fusion module is used to dynamically generate adaptive sampling kernels for each position in the feature map, enabling the model to focus more on the details of local features while locating the manipulated area. Experimental results on multiple datasets show that this method not only effectively improves the accuracy of image manipulation localization but also enhances the robustness of the model.

Keywords: Image manipulation localization; Content awareness; Dual branch; Hybrid convolution; U-Net

Online publication:

1. Introduction

In recent years, with the rapid development of smart devices, digital images have become widely accessible and easily disseminated. Concurrently, the operational barriers for image editing software have progressively lowered, enabling more individuals to acquire image editing skills. This has made distinguishing manipulated images increasingly challenging. Particularly driven by the recent surge in AIGC technology, image editing tasks can now be accomplished with just a few well-crafted instructions. The emergence of these edited images poses a serious threat to the authenticity and reliability of digital imagery [1]. The illicit applications of manipulated images are extensive, frequently exploited in critical domains such as spreading fake news, fabricating judicial evidence, and infringing intellectual property rights, severely impacting socioeconomic development [2,3]. As a vital method in

image forensics, image tampering localization aims to identify and pinpoint altered regions within images, a task attracting increasing attention from researchers.

Existing image tampering localization methods primarily fall into two categories: those based on traditional feature extraction and those based on deep learning. Traditional feature extraction methods employ manually designed features for extraction, perform statistical analysis on the extracted features, and use this to determine image tampering and its location. Traditional feature extraction methods mainly include methods based on color filter array (CFA) [4,5] consistency detection methods, which analyze the damage caused by tampering operations to the CFA interpolation patterns specific to different camera models to locate tampered regions. Methods based on illumination consistency [6–8] analyze inconsistencies in the direction, intensity, and shadow distribution of illumination across different object surfaces within an image to locate tampered regions. Detection methods based on imaging system noise consistency [9–11] detect tampering by identifying differences in systematic noise distribution (e.g., sensor noise) between tampered regions and the overall image. Methods based on JPEG compression artifact consistency [12–14] identify tampered regions through anomalies in JPEG compression artifacts introduced by tampering (e.g., inconsistent block effects, double compression artifacts). These approaches offer high interpretability but detect only limited types of tampering, with constrained localization accuracy and robustness in complex scenarios.

Leveraging the powerful feature extraction capabilities of deep learning, particularly in recent years, an increasing number of researchers have applied it to the field of image tampering, gradually developing deep learning-based image tampering detection methods. Based on mainstream deep learning network architectures, these methods fall into two categories: those based on convolutional neural networks [15–19] and those based on transformers [20–22]. Convolutional neural network-based methods utilize convolutions to extract local image features, employing upsampling or deconvolution operations to achieve pixel-level localization. Transformer-based methods enhance the model's focus on tampered regions by incorporating attention mechanisms and integrating features from different scales [23,24].

Although existing methods based on convolutional neural networks have significantly improved the accuracy of image tampering localization, some shortcomings remain. Current approaches suffer from limited receptive fields due to the size constraints of neural network convolutional kernels. Furthermore, during feature propagation, high-frequency information related to tampering traces is prone to loss. Cross-scale feature interactions are weak, and feature fusion across different levels is insufficient. Furthermore, existing methods employ fixed sampling kernels during feature fusion, failing to effectively capture local details and global contextual information. This inability to focus on local feature variations limits localization accuracy.

To address the aforementioned issues, this paper proposes an image tampering localization method based on dual-branch hybrid convolutions. First, a dual-branch hybrid convolution module is designed. The spatial domain feature extraction branch employs two dilation convolutions to expand the receptive field, enhancing the capture of multi-scale feature information. The frequency domain detail enhancement branch utilizes WTConv to more effectively improve the model's extraction of frequency feature information. This dual-branch hybrid convolution module enables the model to extract multi-scale feature information from both spatial and frequency domains, enhancing its ability to detect tampering traces and thereby improving localization accuracy. Second, introducing content-aware upscaling, we designed a multi-scale content-aware feature fusion module. This module dynamically generates adaptive sampling kernels for each feature map through multi-scale content-aware upscaling, enabling the fused features to effectively capture contextual information and local detail features at

different scales within the tampering region.

2. Research methods

In image manipulation localization tasks, the location is typically determined based on pixel differences between the manipulated and authentic regions. This is primarily achieved by extracting feature information from the manipulated image, learning edge features, frequency domain features, artifact features, and other characteristic details. Leveraging the powerful feature extraction capabilities of deep learning, the manipulated regions within the image are localized. To enhance the accuracy of manipulation localization, it is necessary to strengthen the model's ability to extract these manipulation-specific features.

2.1. Model architecture

This paper proposes an image tampering localization method based on dual-branch hybrid convolutions. The proposed method adopts RRU-Net [25] as its backbone network, primarily consisting of an encoder and a decoder. The encoder performs feature extraction and downsampling on the input image through dual-branch hybrid convolutions and max-pooling layers, while the decoder fuses features via content-aware upsampling. Additionally, the recurrent residual structure within this network addresses the vanishing gradient problem caused by excessive network depth, enhancing the utilization of contextual spatial information in images. The specific network architecture is illustrated in **Figure 1**.

First, the input image $X \in \mathbb{R}^{H \times W \times C}$ is fed into the encoder on the left. The feature maps obtained from each encoding layer are downsampled and used as input for the next encoding layer. Ultimately, the encoder produces feature outputs at different scales.

$$F_i = D_i(X_{i-1}), i \in \{1,2\ 3\ 4\ 5\} \quad F_i = \{1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}\} \tag{1}$$

Here, D_i denotes the encoder. Subsequently, the features extracted by the encoder are fused with those processed through multiscale content-aware upscaling, enabling the fused features to effectively utilize contextual information and variations in local features.

$$U_i = upsample_i(U_{i+1}, F_i), i \in \{1, 2, 3\} \tag{2}$$

$$U_4 = upsample_4(F_5, F_4) \tag{3}$$

Finally, after a 1×1 convolution, the predicted tampering region mask map $Y = Conv(U_1) \in \mathbb{R}^{H \times W \times 1}$ is obtained.

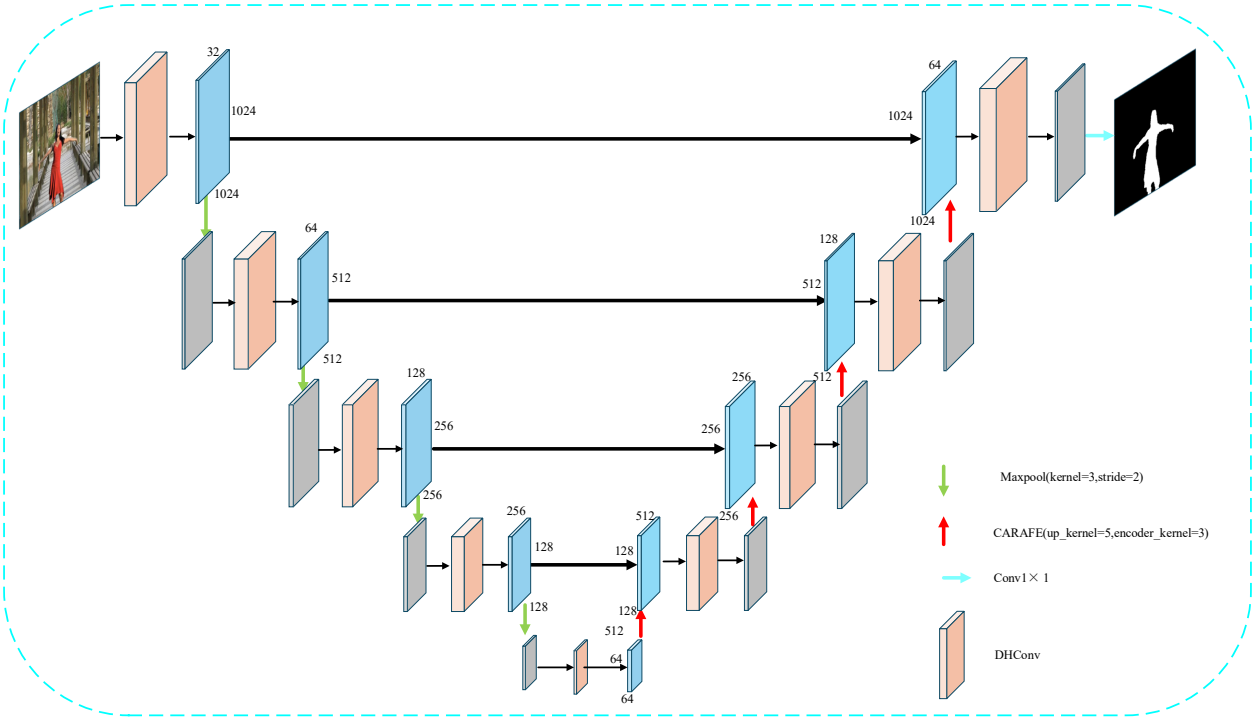


Figure 1. Overall network architecture

2.2. Dual-branch hybrid convolution

Image manipulation operations typically leave traces of human intervention in the altered regions, and these traces constitute crucial features that can be learned by manipulation localization methods. Feature extraction at a single scale captures extremely limited information and fails to fully leverage the complementary nature of multi-scale data. Consequently, in recent years, an increasing number of researchers have turned to multi-scale features for image manipulation localization studies.

In this paper, to leverage frequency-domain information across different scales while addressing limitations such as restricted receptive fields in traditional convolutions and increased computational overhead and model complexity when expanding receptive fields using conventional methods, we designed a dual-branch hybrid convolutional module inspired by Finder *et al.* [26] to enhance the model's feature learning capability. As shown in **Figure 1**, within the encoder, the dual-branch hybrid convolution module extracts image features at different scales. Low-scale features are fed into the high-scale feature extraction process, ultimately yielding a multi-scale feature map $\{F_1, F_2^{\frac{1}{2}}, F_3^{\frac{1}{4}}, F_4^{\frac{1}{8}}, F_5^{\frac{1}{16}}\}$. In the decoder, high-scale feature maps undergo successive upsampling, with the dual-branch hybrid convolution module extracting image features at varying scales. The structure of the dual-branch hybrid convolution module is illustrated in **Figure 2**.

As shown in **Figure 2**, the dual-branch hybrid convolution module is primarily divided into two branches: the spatial receptive field expansion branch, mainly composed of dilated convolutions, extracts spatial domain features; the frequency domain detail enhancement branch, primarily composed of wavelet convolutions, extracts frequency domain features.

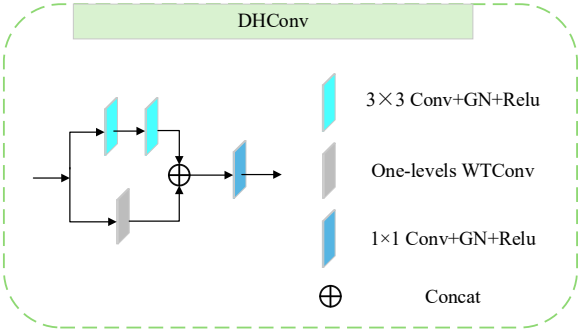


Figure 2. Dual-branch hybrid convolution module

In the spatial receptive field expansion branch, two 3×3 dilated convolutions with a dilation rate of 2 and ReLU activation functions are applied to the input feature map to expand the model's receptive field. This enhances the model's ability to extract contextual semantic information, thereby capturing the spatial feature information X_s of the input features. In the frequency domain detail enhancement branch, WTConv wavelet transformation is employed to separate low-frequency and high-frequency information from the input feature map. This wavelet transform employs four distinct filters: the *LL* filter captures low-frequency information, the *LH* filter captures horizontal information, the *HL* filter captures vertical information, and the *HH* filter captures diagonal information. Among these, *LL* is a low-pass filter, while *LH*, *HL*, and *HH* form a set of high-pass filters. These four filters constitute an orthogonal basis as shown in (4):

$$f_{LL}=\frac{1}{2}\begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix},\ f_{LH}=\frac{1}{2}\begin{bmatrix}1 & -1\\ 1 & -1\end{bmatrix},f_{HL}=\frac{1}{2}\begin{bmatrix}1 & 1\\ -1 & -1\end{bmatrix},f_{HH}=\frac{1}{2}\begin{bmatrix}1 & -1\\ -1 & 1\end{bmatrix} \tag{4}$$

For each input channel, the convolution output has four channels, each with a resolution equal to half of *X*.

$$[X_{LL},X_{LH},X_{HL},X_{HH}]=Conv5\times5([f_{LL},f_{LH},f_{HL},f_{HH}],X) \tag{5}$$

Among these, convolving the orthogonal basis generated by the four filters with 5×5 produces the low-frequency component X_{LL} of the input *X*, along with the horizontal, vertical, and diagonal high-frequency components X_{LH}, X_{HL}, and X_{HH}.

For the orthogonal basis of equation (4), the inverse wavelet transform (IWT) is obtained through transpose convolution:

$$X_I=transposedConv([f_{LL},f_{LH},f_{HL},f_{HH}],[X_{LL},X_{LH},X_{HL},X_{HH}]) \tag{6}$$

By separating the convolution operation from the frequency components through wavelet transformation, high-frequency and low-frequency information is isolated. The enhanced low-frequency information improves feature extraction. Ultimately, the frequency domain detail enhancement branch extracts the frequency feature information X_F from the input *X*.

$$X_F=Conv(X)+IWT(X_I) \tag{7}$$

The WTConv wavelet transform structure is shown in **Figure 3**.

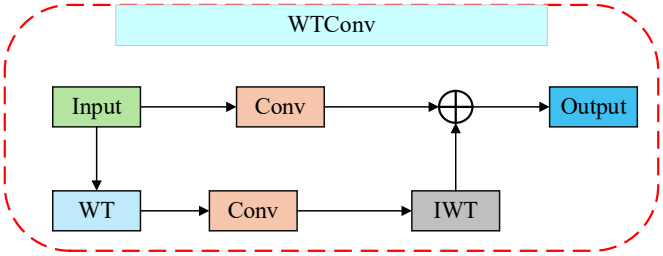


Figure 3. WTConv Architecture

Finally, the spatial feature information X_S extracted by the spatial feature field expansion branch and the frequency feature information X_F extracted by the frequency detail enhancement branch are fused. This fusion undergoes a 1×1 convolution to produce the final output features.

2.3. Multi-scale content-aware feature fusion

In our approach, to fully leverage feature information across different scales and effectively capture both contextual information and local detail features of the tampered region at various scales, we designed a multi-scale content-aware feature fusion module at the decoder end, inspired by Wang *et al.* [27]. As shown in **Figure 1**, at the decoder stage, high-scale features undergo content-aware upsampling and are concatenated with low-scale features. After four rounds of upsampling, the fused features are obtained. These fused features effectively perceive the contextual information and local detail features of the tampered region across multiple scales, enhancing the model's ability to recognize tampering edges and texture details. The network architecture for content-aware upsampling is illustrated in **Figure 4**.

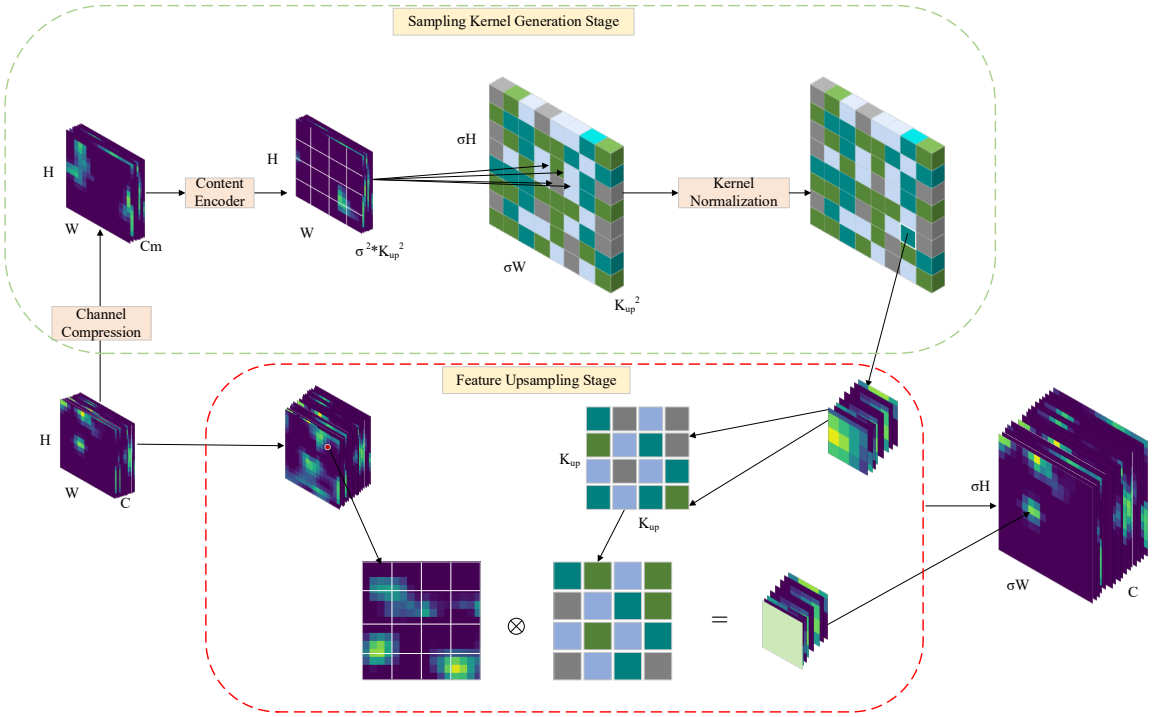


Figure 4. Content-aware upsampling network architecture

As shown in **Figure 4**, the content-aware upscaling process primarily consists of two stages: the sampling

kernel generation stage and the feature upscaling stage. During the sampling kernel generation stage, adaptive sampling kernels are dynamically generated for each position of the target feature based on its specific content. The specific process is as follows: For the input feature map $F \in \mathbb{R}^{H\times W\times C}$, a 1×1 convolution is first applied to perform channel compression on feature F, yielding the compressed feature map $F' \in \mathbb{R}^{H\times W\times C_m}$, where C_m denotes the number of compressed channels. Then, encoding is performed using the content encoder to obtain $F'' \in \mathbb{R}^{H\times W\times(\sigma^2\times k_{up}^2)}$. This encoder employs a 3×3 convolution kernel, where σ denotes the upsampling factor and k_{up} represents the upsampling kernel size. Subsequently, F'' undergoes pixel reordering to produce $F''' \in \mathbb{R}^{\sigma H\times\sigma W\times k_{up}^2}$. Finally, F''' is normalized to generate the sampling kernel F_{kernel}, creating a sampling kernel $k_{up}\times k_{up}$ with k_{up}^2 weights for each position.

During the feature sampling phase, the input features are upsampled based on the generated sampling kernels. First, for each position $p = (i,j)$ in the input feature $F \in \mathbb{R}^{H\times W\times C}$, extract the $k_{up}\times k_{up}$ neighborhood N_p centered at $p = (i,j)$. Then, at the corresponding position $p' = (i',j')$ in the sampling kernel F_{kernel} generated during the kernel sampling phase, the corresponding $k_{up}\times k_{up}$ sampling kernel $K_{p'}$ is extracted. Multiplying the corresponding positions of F and $K_{p'}$ and summing the results yields the feature value at that position for the target feature. Performing this operation for each position produces the output feature map $F_{out} \in \mathbb{R}^{\sigma H\times\sigma W\times C}$.

3. Experimental design and results

3.1. Experimental setup

(1) Dataset: To validate the effectiveness of our method, we conducted experiments on various public datasets, including CASIAv1 [28], CASIAv2 [28], Columbia [29], and NIST16 [30]. To ensure a more scientific and accurate comparison, we split the Columbia and NIST16 datasets into training and testing sets. For CASIA, we used CASIAv2 as the training set and CASIAv1 as the testing set for experiments. The dataset configurations are shown in **Table 1**. All training images underwent data augmentation techniques such as rotation, cropping, and transformation.

Table 1. Training and test set partitioning of the dataset

Datasets	CASIA	Columbia	NIST16
Train	5123	126	404
Test	921	54	160

(2) Experimental environment: The experiment was implemented using the PyTorch deep learning framework, with Python 3.8 as the programming environment. Training was conducted on an NVIDIA GeForce RTX 3090 GPU. During training, the Adam optimizer was employed with an initial learning rate of 1e-4 and 200 epochs. Input image resolutions were adjusted based on resolution characteristics for different training datasets.

(3) Evaluation metrics: This paper primarily employs pixel-level F1 score and area under the curve (AUC) as model evaluation metrics. These represent the two most commonly used core indicators for assessing model performance in image tampering localization tasks, measuring the model's localization accuracy and discrimination capability from different perspectives, respectively. They serve complementary evaluation functions within this research domain, collectively establishing a multidimensional

performance assessment framework.

The Pixel-level F1 Score is the harmonic mean of Precision and Recall, used to evaluate a model's accuracy in locating tampered regions at the pixel level. It is defined as:

$$F_1 = \frac{2Precision * Recall}{Precision + Recall} \tag{8}$$

Among these, accuracy is defined as:

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

The recall rate is defined as:

$$Recall = \frac{TP}{TP + FN} \tag{10}$$

The AUC evaluates a model's overall performance across different resolution thresholds by calculating the area enclosed by the receiver operating characteristic (ROC) curve. The ROC curve is generated by adjusting classification thresholds, with true positive rate (TPR) plotted on the vertical axis and false positive rate (FPR) on the horizontal axis. The closer the AUC value is to 1, the better the model performs in classifying pixels between the tampered region and the true background.

3.2. Ablation experiment

To validate the effectiveness of the proposed dual-branch hybrid convolution and multi-scale content-aware feature fusion module, ablation experiments were designed on the CASIAv1, Columbia, and NIST16 datasets. The experimental design is as follows:

Option 1: Remove all modules simultaneously to validate the baseline model's performance across datasets;

Option 2: Remove the dual-branch hybrid convolution module and use standard convolutions for feature extraction within the network;

Option 3: Remove the multi-scale content-aware feature fusion module and directly up-sample the encoder's output feature map to the decoder for prediction;

Option 4: Utilize all modules simultaneously to validate the model's overall performance across datasets.

As shown in the ablation experiment results in **Table 2**, when all modules are used simultaneously, the model achieves an average F1 score of 0.691 and an average AUC of 0.871 across all datasets. Removing any single module results in a decrease in both the average F1 score and average AUC. The baseline model exhibits the lowest average F1 score and average AUC, validating the effectiveness of the proposed modules. In Scheme 3, removing the multi-scale content-aware feature fusion module resulted in a significant decrease in both F1 and AUC scores, dropping by 4.8 percentage points and 3.7 percentage points, respectively. This indicates that multi-scale content-aware feature fusion effectively captures contextual information and local detail features across different scales, thereby enhancing localization accuracy.

Table 2. Ablation experiment results

Option	CASIAv1		Columbia		NIST16		MEAN	
	F1	AUC	F1	AUC	F1	AUC	F1	AUC
1	0.397	0.728	0.724	0.839	0.850	0.985	0.657	0.851
2	0.411	0.742	0.745	0.869	0.896	0.992	0.684	0.868
3	0.417	0.751	0.640	0.766	0.873	0.985	0.643	0.834
4	0.422	0.751	0.750	0.869	0.900	0.993	0.691	0.871

3.3. Objective evaluation

To objectively evaluate model performance, the proposed method is compared with state-of-the-art approaches, including SPAN [18], GSR-Net [31], DenseFCN [32], LocateNet [33], RGB-N [15], U-Net [24], and RRU-Net [25]. To assess model effectiveness, we recorded the F1 scores of each method across different datasets. Additionally, to visually compare the overall performance of different approaches, we calculated the average F1 scores for each method across various datasets. The results are presented in **Table 3**.

Table 3. Comparison of F1 scores across different methods on each dataset

Method	F1 scores (%)			
	CASIAv1	Columbia	NIST6	MEAN
RGB-N [15]	40.8	69.7	72.2	60.9
SPAN [18]	38.2	81.5	58.2	48.0
LocateNet [33]	27.3	41.1	73.8	47.4
GSRNet [31]	34.0	43.3	64.0	47.1
DenseFCN [32]	20.3	25.7	81.2	42.4
U-Net [24]	35.0	51.5	67.2	51.2
RRU-Net [25]	39.7	72.4	85.0	65.7
Ours	42.2	75.0	90.0	69.1

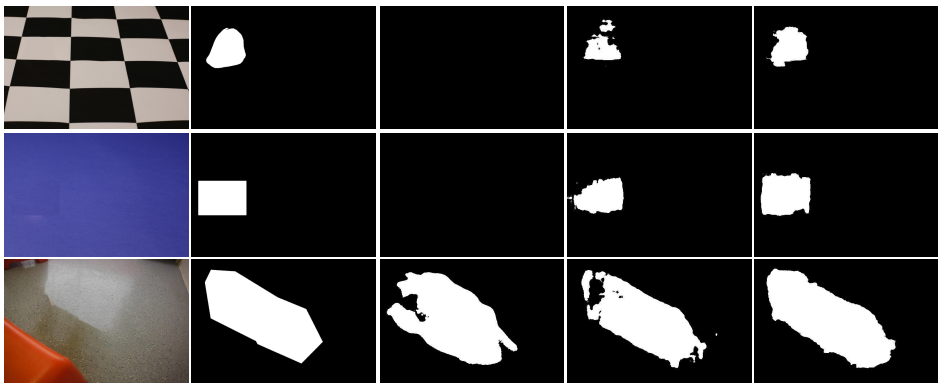
As shown in **Table 3**, the model achieves an average F1 score of 69.1% across the CASIAv1, Columbia, and NIST16 datasets, representing a 3.4 percentage point improvement over RRU-Net. Our method achieves the highest F1 scores on both the CASIAv1 and NIST16 datasets. On the Columbia dataset, it falls below the SPAN method. This discrepancy stems from the Columbia dataset's homogeneous tampering types and high image resolution. During the adjustment of input image resolution, the tampered regions may have been adversely affected.

3.4. Subjective evaluation

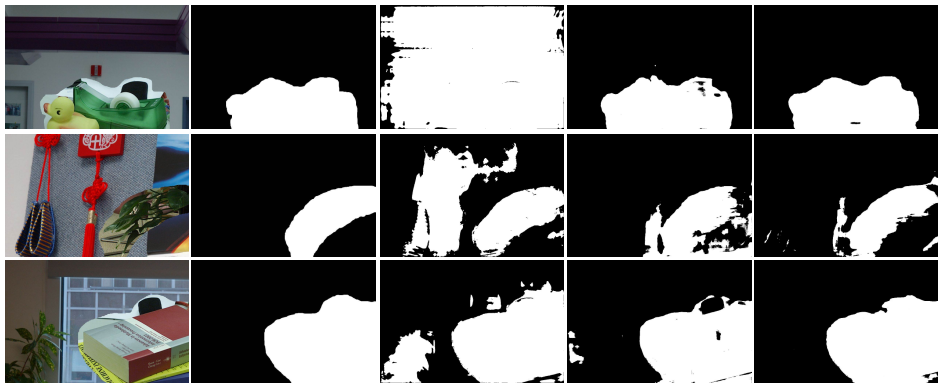
To enable a more intuitive comparison from a subjective visual perspective, the proposed method is compared with other approaches based on their predicted masks on the test dataset. By contrasting the predicted mask images with the ground truth masks, differences in localization accuracy, boundary clarity, and false detection rate among various methods can be clearly observed. Selected results are shown in **Figure 5**. **Figure 5(a)** shows the comparison of predicted masks from different methods on the CASIA v1 dataset, **Figure 5(b)** shows the

comparison on the NIST16 dataset, and **Figure 5(c)** shows the comparison on the Columbia dataset. In these three figures, Column 1 shows the tampered image. Column 2 displays the true mask image. Column 3 presents the predicted mask from the U-Net method. Column 4 shows the predicted mask from RRU-Net. Column 5 displays the predicted mask from the proposed method.

(a) Comparison of prediction masks across different methods on the CASIA v1 dataset



(b) Comparison of prediction masks across different methods on the NIST16 dataset



(c) Comparison of prediction masks across different methods on the Columbia dataset

Figure 5. Comparison of prediction masks between the proposed method and other methods

Figure 5 demonstrates that compared to other methods, our approach achieves more precise localization of tampered regions, with clearer predicted boundaries and higher overlap with the ground truth mask. Additionally, our method exhibits favorable false positive rates, significantly reducing instances where pixels from genuine

regions are erroneously identified as tampered. This indicates strong capability in distinguishing between authentic and tampered image regions.

3.5. Robustness assessment

In image tampering localization tasks, robustness evaluation is a critical step for verifying model effectiveness and reliability. Real-world images may be subject to various disturbances such as noise, light pollution, compression, and transformations, which significantly impact model performance. Robustness evaluation tests a model's stability under different disturbance conditions and validates its generalization capability.

In this paper, we conduct robustness experiments on the CASIAv1 dataset, applying two types of image distortion: Gaussian Blur and JPEG Compression. The experimental results are shown in **Figure 6**. As illustrated in **Figure 6**, the proposed method demonstrates strong robustness against both types of interference attacks.

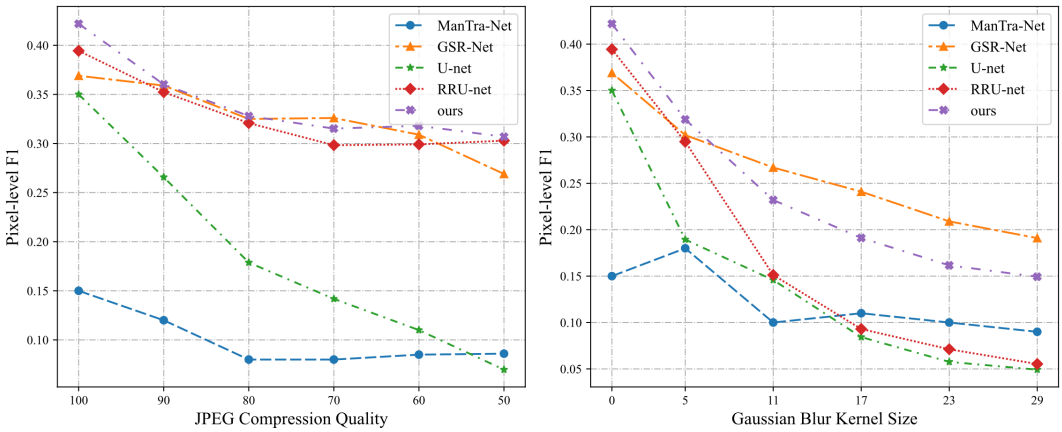


Figure 6. Robustness evaluation of the model on the CASIAv1 dataset

4. Conclusion

This paper proposes an image tampering localization method based on dual-branch hybrid convolutions. First, a dual-branch hybrid convolution module is designed to expand the model's receptive field and enhance its ability to extract contextual semantic information. This enables the model to focus on high-frequency detail features of tampering traces while locating tampered regions. Second, a multi-scale content-aware feature fusion module is employed to dynamically generate adaptive sampling kernels for features at different scales. This enables the model to identify tampering edges and local feature details while locating tampered regions, effectively enhancing robustness and generalization capabilities. However, challenges remain: the proposed method's localization accuracy requires improvement for datasets with limited tampering types and low image resolution. Future research could explore incorporating edge information during feature extraction to further enhance the model's localization precision.

Funding

National Natural Science Foundation of China (61703363); Shanxi Provincial Basic Research Program (202403021221206); Key Project of Shanxi Provincial Strategic Research on Science and Technology

(202304031401011); Funding Project for Scientific Research Innovation Team on Data Mining and Industrial Intelligence Applications (YCXYTD-202402); Yuncheng University Research Project (YQ-2020021)

Disclosure statement

The authors declare no conflict of interest.

References

[1] Jin X, Yu W, Shi W, 2024, Image Manipulation Localization via Dynamic Cross-Modality Fusion and Progressive Integration. Neurocomputing, 610: 128607.

[2] Wei H, Yan C, Li H, 2024, Image Tampering Localization Based on Integrated Multiscale Attention. Journal of Computer-Aided Design & Computer Graphics, 36(08): 1237–1245.

[3] Zeng Z, Tan P, 2025, Image Tampering Detection and Localization Model Based on Multi-Branch HRNet. Modern Electronic Technique, 48(03): 35–42.

[4] Varlamova AA, Kuznetsov AV, 2017, Image Splicing Localization Based on CFA-Artifacts Analysis. Computer Optics, 41(6): 920–930.

[5] Hussien NY, Mahmoud RO, Zayed HH, 2020, Deep Learning on Digital Image Splicing Detection Using CFA Artifacts. International Journal of Sociotechnology and Knowledge Development (IJSKD), 12(2): 31–44.

[6] Vidyadharan DS, Thampi SM, 2018, Evaluating Color and Texture Features for Forgery Localization from Illuminant Maps. Multimedia Tools and Applications, 77: 21131–21161.

[7] Niyishaka P, Bhagvati C, 2021, Image Splicing Detection Technique Based on Illumination-Reflectance Model and LBP. Multimedia Tools and Applications, 80(2): 2161–2175.

[8] Zhe S, Peng S, 2020, Authentication of Splicing Manipulation by Exposing Inconsistency in Color Shift. Multimedia Tools and Applications, 79(11): 8235–8248.

[9] Lyu S, Pan X, Zhang X, 2014, Exposing Region Splicing Forgeries with Blind Local Noise Estimation. International Journal of Computer Vision, 110: 202–221.

[10] Dong J, Chen L, Tian J, et al., 2016, A Novel Image Splicing Detection Method Based on the Inconsistency of Image Noise, 2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA), IEEE, 560–563.

[11] Zhu N, Li Z, 2018, Blind Image Splicing Detection via Noise Level Function. Signal Processing: Image Communication, 68: 181–192.

[12] Wang SL, Liew AWC, Li SH, et al., 2014, Detection of Shifted Double JPEG Compression by an Adaptive DCT Coefficient Model. EURASIP Journal on Advances in Signal Processing, 2014: 1–17.

[13] Thai TH, Cogranne R, Retraint F, et al., 2016, JPEG Quantization Step Estimation and Its Applications to Digital Image Forensics. IEEE Transactions on Information Forensics and Security, 12(1): 123–133.

[14] Iakovidou C, Zampoglou M, Papadopoulos S, et al., 2018, Content-Aware Detection of JPEG Grid Inconsistencies for Intuitive Image Forensics. Journal of Visual Communication and Image Representation, 54: 155–170.

[15] Zhou P, Han X, Morariu VI, et al., 2018, Learning Rich Features for Image Manipulation Detection, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1053–1061.

[16] Wu Y, Abd Almageed W, Natarajan P, 2019, Mantra-Net: Manipulation Tracing Network for Detection and Localization of Image Forgeries with Anomalous Features, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 9543–9552.

[17] Bayar B, Stamm MC, 2018, Constrained Convolutional Neural Networks: A New Approach Towards General Purpose Image Manipulation Detection. IEEE Transactions on Information Forensics and Security, 13(11): 2691–2706.

[18] Hu X, Zhang Z, Jiang Z, et al., 2020, SPAN: Spatial Pyramid Attention Network for Image Manipulation Localization, Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16, Springer International Publishing, 312–328.

[19] Chen X, Dong C, Ji J, et al., 2021, Image Manipulation Detection by Multi-View Multi-Scale Supervision, Proceedings of the IEEE/CVF International Conference on Computer Vision, 14185–14193.

[20] Hao J, Zhang Z, Yang S, et al., 2021, Transforensics: Image Forgery Localization with Dense Self-Attention, Proceedings of the IEEE/CVF International Conference on Computer Vision, 15055–15064.

[21] Wang J, Wu Z, Chen J, et al., 2022, Objectformer for Image Manipulation Detection and Localization, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2364–2373.

[22] Ma X, Du B, Jiang Z, et al., 2023, IML-ViT: Benchmarking Image Manipulation Localization by Vision Transformer. arXiv. https://arxiv.org/abs/2307.14863

[23] Zeng K, Cheng R, Tan W, et al., 2024, MGQFormer: Mask-Guided Query-Based Transformer for Image Manipulation Localization, Proceedings of the AAAI Conference on Artificial Intelligence, 38(7): 6944–6952.

[24] Ronneberger O, Fischer P, Brox T, 2015, U-Net: Convolutional Networks for Biomedical Image Segmentation, Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III 18, Springer International Publishing, 234–241.

[25] Bi X, Wei Y, Xiao B, et al., 2019, RRU-Net: The Ringed Residual U-Net for Image Splicing Forgery Detection, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops.

[26] Finder SE, Amoyal R, Treister E, et al., 2024, Wavelet Convolutions for Large Receptive Fields, European Conference on Computer Vision, Springer Nature Switzerland, Cham, 363–380.

[27] Wang J, Chen K, Xu R, et al., 2019, Carafe: Content-Aware Reassembly of Features, Proceedings of the IEEE/CVF International Conference on Computer Vision, 3007–3016.

[28] Dong J, Wang W, Tan T, 2013, Casia Image Tampering Detection Evaluation Database, 2013 IEEE China Summit and International Conference on Signal and Information Processing, IEEE, 422–426.

[29] Hsu YF, Chang SF, 2006, Detecting Image Splicing Using Geometry Invariants and Camera Characteristics Consistency, 2006 IEEE International Conference on Multimedia and Expo, IEEE, 549–552.

[30] Guan H, Kozak M, Robertson E, et al., 2019, MFC Datasets: Large-Scale Benchmark Datasets for Media Forensic Challenge Evaluation, 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW), IEEE, 63–72□

[31] Zhou P, Chen B C, Han X, et al., 2020, Generate, Segment, and Refine: Towards Generic Manipulation Segmentation, Proceedings of the AAAI Conference on Artificial Intelligence, 34(07): 13058–13065.

[32] Zhuang P, Li H, Tan S, et al., 2021, Image Tampering Localization Using a Dense Fully Convolutional Network. IEEE Transactions on Information Forensics and Security, 16: 2986–2999.

[33] Zhuo L, Tan S, Li B, et al., 2022, Self-Adversarial Training Incorporating Forgery Attention for Image Forgery Localization. IEEE Transactions on Information Forensics and Security, 17: 819–834.