MAMQ-Net: Robust Image Forgery Localization via Complementary Tampering Features and Multi-level Tamper-aware Queries Aggregation

Minghao Jia[1], **Zhuoyi Zhang**[1], **Jiwei Zhang**[1*], **Siwei Wang**[2], **Feifei Kou**[1], **Lei Shi**[3], **Shaozhang Niu**[1], **Jiaming Pei**[4]

[1]Beijing University of Posts and Telecommunications, Beijing, China
[2]National University of Defense Technology, Changsha, China
[3]Communication University of China, Beijing, China
[4]The University of Sydney, Sydney, Australia

Abstract

Image Forgery Localization (IFL) aims to identify and localize the tampered regions within edited images. Many studies employ a dual-branch backbone to extract tampering features from RGB modality and low-level modality, followed by feature fusion at the final stage. In this process, the extraction and fusion of dual-modality features is relatively independent, which fails to fully leverage the complementarity between different modalities and thus diminishes sensitivity to tampering artifacts. Inspired by the way humans continuously integrate multi-faceted knowledge to understand the world, we propose MAMQ-Net, which contains a novel Multi-stage Alternating Feature Extraction and Interaction architecture. At each stage, we deeply explore the intrinsic relationships and mappings between different modality features. Feature extraction and interaction are performed alternately, constructing complementary dual-modality tampering feature representations and enhancing sensitivity to tampering artifacts. Additionally, we introduce a lightweight, Query-driven Multi-level Feature Decoding. This mechanism progressively aggregates key information from multi-level dual-modality tampering features through multiple sets of learnable tamper-aware queries, effectively filtering out irrelevant features. Finally, multi-level queries are used to refine discriminative features, enabling precise localization of tampered regions. Extensive experiments demonstrate that our framework outperforms current state-of-the-art models in localization accuracy and robustness across multiple public datasets, achieving a favorable balance between performance and efficiency. Code is available at `https://github.com/jmhppj/MAMQ-Net`.

1 Introduction

The widespread use of digital image tampering techniques poses a severe challenge to societal trust systems. Tampered images are often used to create fake news, forge evidence, or commit fraud, posing serious threats to personal privacy, social order, and even national security. IFL has become a critical technological barrier for maintaining information authenticity.

Most existing IFL frameworks [1][2][3][4] typically adopt a dual-branch structure, where one branch extracts high-level features from the RGB modality, and the other extracts low-level features from noise or high-frequency modality, working together to locate tampered regions. These frameworks often follow the classical paradigm of "extraction $\rightarrow$ simple fusion" (as shown in Fig. 2 (a)). This

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

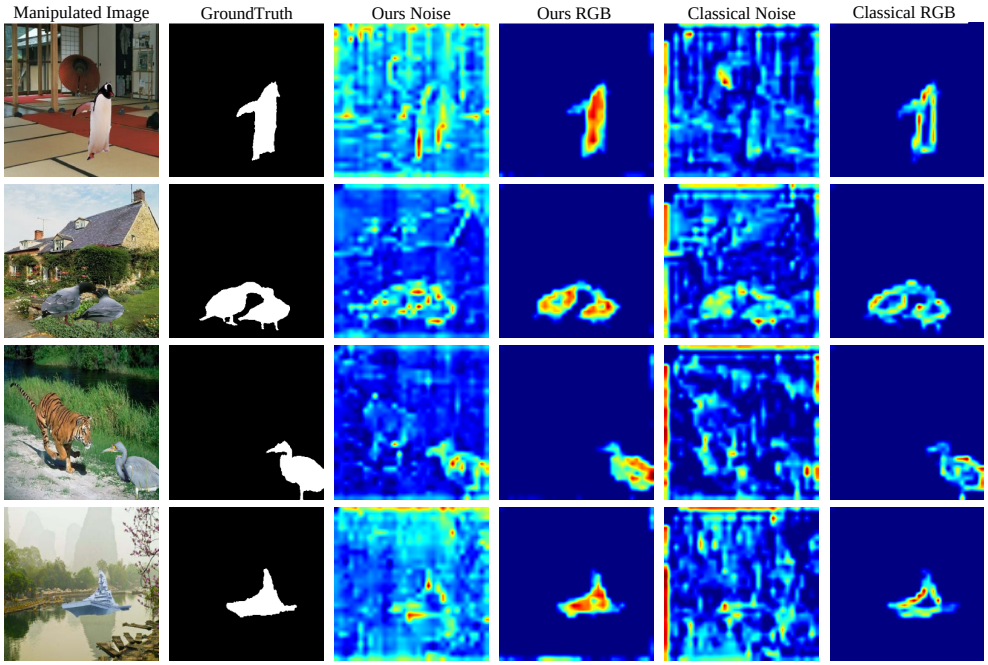


Figure 1: Ours vs. Classical. Grad-CAM visualizations of backbone output features under different paradigms. The redder the color of the region, the greater its contribution to the model's prediction results.

paradigm suffers from severe modality isolation and information fragmentation, which results in a lack or insufficiency of interaction among features at early and intermediate levels. For example, the RGB modality focuses on semantic boundaries but cannot perceive local statistical anomalies present in the noise modality; conversely, the noise modality is sensitive to compression artifacts but lacks global semantic guidance, often misclassifying highly textured regions (e.g., leaves) as tampered. Therefore, we hope to design a comprehensive bidirectional inter-modal interaction mechanism to construct complementary dual-modality tampering feature representations, thereby enhancing the sensitivity of dual-modality features to tampering artifacts. This mechanism follows a cyclic paradigm of "extract $\rightarrow$ interact $\rightarrow$ feedback", as shown in Fig. 2 (b). Through experiments, we visualize the features output by the backbone under both classical paradigm (Columns 5 and 6) and our methods (Columns 3 and 4) in Fig. 1. It can be observed that dual-modality features under our methods exhibit higher sensitivity to tampered regions.

Current multi-level feature decoding methods often employ upsampling combined with convolution or MLP [5][6], which indiscriminately aggregate a large amount of irrelevant background information and noise, severely overwhelming subtle tampering features. Moreover, fixed or simple decoding strategies struggle to adaptively balance the contributions of features from different levels and modalities, preventing effective collaboration between shallow fine-grained textures and deep semantic context, and thus degrading localization accuracy. In addition, computationally intensive decoding hinders their deployment in resource-constrained scenarios. Therefore, we decide to design the decoding mechanism of our framework to be lightweight and capable of effectively screening and aggregating critical information from multi-level features.

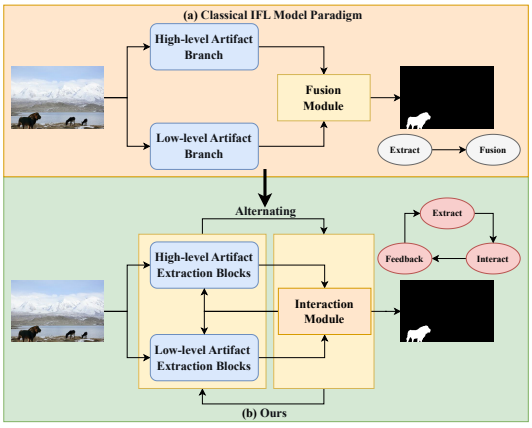


Figure 2: Differences between the classical IFL model paradigm and our method.

Recent studies have introduced multimodal large models into the task of IFL [7][8][9], using them as a backbone to extract general features. While this approach enables the extraction of more comprehensive tampering features, it also introduces a large amount of irrelevant information [10]. Without sufficient feature selection, this can easily lead to counterproductive results. Moreover, multimodal large models have more critical limitations, including the need for substantial computational resources and slower inference speed, making deployment on mobile devices challenging. Our model aims to incorporate as few additional parameters as possible based on a relatively lightweight backbone, achieving a favorable balance between performance and efficiency.

Motivated by the above observations, we propose MAMQ-Net, which contains a Multi-stage Alternating Feature Extraction and Interaction architecture and a lightweight, Query-driven Multi-level Feature Decoding. The former is inspired by the human cognitive process of repeatedly examining complex objects from multiple perspectives to gain a deep understanding. At each stage, feature extraction and deep interaction are not isolated but performed alternately and mutually reinforced. Through specially designed cross-modal Feature Alignment and Dual-modal Feature Cross-guided Module, the network bidirectionally and deeply explores previously underutilized intrinsic relationships and mappings between RGB and noise modality features at every stage. This process essentially enables both modalities to perform bidirectional retrieval and attention across multiple levels, perceiving important relevant points within each modality while suppressing irrelevant noise. To allow such interaction to propagate across levels, the interacted features are fed back into the backbone to extract the next-level features. In this way, we successfully construct complementary dual-modality tampering feature representations, significantly enhancing sensitivity to tampering artifacts. Furthermore, we innovatively introduce a lightweight decoding mechanism. By incorporating multiple sets of learnable, dedicated tamper-aware query vectors, these queries act as “fusion controllers" for each feature level. At each stage, through a carefully designed Multi-domain Feature Aggregation Module, the queries progressively and selectively extract and condense the most critical information from dual-modality features while filtering out irrelevant interference. This process effectively simulates a coarse-to-fine, stepwise focusing decision procedure: shallow-level queries target potential anomalous regions, whereas deep-level queries associate global context to verify and refine localization results. Finally, the multi-level dual-modality features and their corresponding queries are fed into a Query-driven Multi-level Feature Decoder to localize the tampered regions. Such a dynamic, decision-level fusion strategy substantially improves localization accuracy.

In summary, our main contributions are as follows:

- We propose MAMQ-Net, a novel framework with a Multi-stage Alternating Feature Extraction and Interaction architecture. At each stage, cross-modal Feature Alignment and a Dual-modal Feature Cross-guided Module are employed to construct complementary dual-modality tampering representations, significantly enhancing sensitivity to tampering artifacts.
- We propose a lightweight, Query-driven Multi-level Feature Decoding. Multiple sets of learnable tamper-aware queries progressively aggregate key dual-modality features at each stage via a Multi-domain Feature Aggregation Module while filtering out irrelevant information. These queries act as “fusion controllers” for decision-level feature selection and fusion, substantially improving localization accuracy.
- Extensive experiments show that our framework outperforms state-of-the-art models in localization accuracy and robustness on multiple public datasets, achieving an optimal trade-off between performance and complexity.

2 Related work

2.1 Image Forgery Localization

Traditional IFL methods mainly rely on hand-crafted extractors [11][12][13][14] to capture anomalies caused by tampering operations. For example, Ferrara et al. utilize CFA [11] pattern inconsistencies to detect forged regions. However, they rely on the assumption that specific tampering operations always leave particular traces. When this assumption is not valid, their performance deteriorates significantly. With the development of computational power, deep learning has achieved remarkable progress in the field of IFL [15][16][17][18]. Wu et al. propose ManTra-Net[19], which formulates

IFL as an anomaly detection problem by modeling 385 manipulation types with Z-score and auxiliary features. Liu et al. introduce PSCC-Net[20], which leverages HRNet[21] to learn multi-scale features and employs SCCM to capture spatial-channel correlations for progressive mask generation. ObjectFormer [22] employs CNN layers to extract local features and then leverages transformer encoders for global modeling. However, they still exhibit limitations in generalization capability and robustness. In the latest studies, researchers have begun exploring multimodal large models (MLM) in IFL to improve generalization and interpretability. For instance, FakeShield [23] utilizes MLM by integrating visual features with linguistic instructions, enabling instruction-driven forgery localization and interpretable outputs. IMDPrompter [24] leverages a Cross-view Prompt Learning paradigm built upon SAM[25] to achieve robust localization. Nevertheless, these approaches often come with higher computational demands, slower inference speed, and introduce irrelevant features.

2.2 Classical paradigm of feature extraction and fusion in IFL

The dual-branch architecture that separately extracts RGB features and high-frequency modality features has become a classic paradigm for IFL. This paradigm typically adopts a sequential feature extraction and simple fusion strategy. For example, MVSS-Net [26] performs feature extraction and fusion in an independent manner, where only the deepest features from ResNet [27] are used for fusion, while shallow features are largely ignored. Such a design limits the model's ability to capture fine-grained cues and hinders comprehensive representation learning. [28] propose the two-stream encoder–decoder network, where two branches independently perform encoding and decoding without cross-branch interaction, thus limiting complementary feature learning. [29] propose an end-to-end IFL network that fuses RGB and noise features at each level using FAM, but without feeding the fused features back to the backbone, which restricts the transmission of fusion information across levels. In summary, such approaches lead to suboptimal representation learning.

2.3 Query-based feature aggregation

Query-based architectures have recently emerged as a powerful paradigm for feature selection in computer vision. The fundamental principle lies in introducing learnable queries that interact with image features through attention mechanisms, thereby steering the network toward task-relevant regions. For example, Mask2Former [30] employs masked cross-attention to refine queries, restricting attention to predicted mask regions and thereby improving segmentation precision. ECENet [31] further generates object queries directly from predicted masks, ensuring that each query is semantically explicit and corresponds to a distinct object region. In SAM [25], output tokens serve as queries that guide image embeddings and prompt embeddings for segmentation. These methods demonstrate the outstanding potential of learnable queries in feature selection and aggregation.

3 Method

3.1 Overall framework of MAMQ-Net

We propose an IFL framework, MAMQ-Net (as shown in Fig. 3), which emulates human learning mechanisms to acquire feature representations sensitive to tampering artifacts. It is primarily composed of two key components: a Multi-stage Alternating Feature Extraction and Interaction, and a Query-driven Multi-level Feature Decoding. The former employs both the RGB view X_{img} and a high-frequency noise view X_{noise} processed by a Multi-domain Noise-sensitive Fusion Module (MNFM) as inputs. The interrelationships between diverse modal features are extensively explored at each stage via a cross-modal Feature Alignment and Dual-modal Feature Cross-guided Module (DFCM). The enhanced features are subsequently fed into the corresponding backbone for next-level feature extraction. This process facilitates the construction of complementary dual-modal feature representations $\{R_1 \sim R_4, N_1 \sim N_4\}$, thereby improving sensitivity to tampering artifacts. The latter employs a set of learnable tamper-aware queries Q_{taq} as input. At each stage, key information from dual-modal tampering features is progressively integrated through a Multi-domain Feature Aggregation Module (MFAM), while irrelevant features are filtered out, resulting in multi-level query embeddings $\{Q_1 \sim Q_4\}$. Finally, the dual-modal features and their corresponding query embeddings at each level form a triplet, which is fed into the Query-driven Multi-level Feature Decoder (QMFD) for decoding, enabling precise localization of tampered regions.

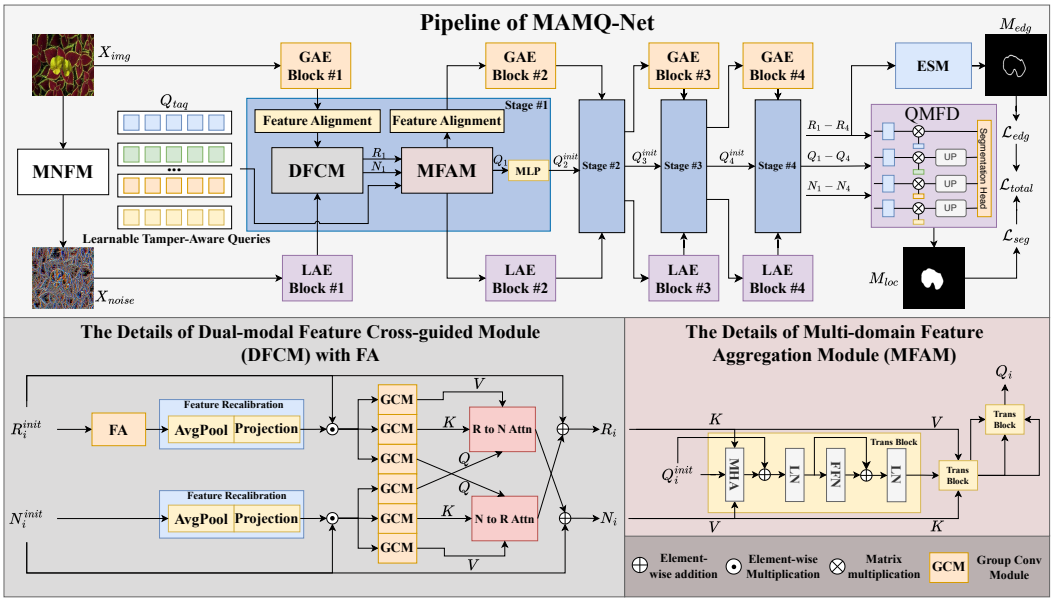


Figure 3: The pipeline of MAMQ-Net. Given an RGB view X_{img} and a high-frequency noise view X_{noise}, they are first processed by the Multi-stage Alternating Feature Extraction and Interaction architecture to generate multi-level complementary feature representations $\{R_1 \sim R_4, N_1 \sim N_4\}$. Subsequently, a set of learnable tamper-aware queries Q_{taq} progressively aggregates cross-modal information through the MFAM. Finally, the tampering features and query embeddings from all levels are fed into the QMFD to localize tampered regions M_{loc}.

3.2 Multi-stage Alternating Feature Extraction and Interaction

Previous methods [2][3] employ a simple approach for feature extraction and fusion. These methods fail to achieve sufficient feature interaction and do not effectively leverage the complementarity between dual-modal features. Accordingly, we propose a Multi-stage Alternating Feature Extraction and Interaction. It follows a continuously "extract→interact→extract→interact" paradigm, where feature extraction and interaction are performed alternately. SegFormer [32] is selected to serve as both the Global and Local Artifact Extractors (GAE and LAE) in the network, and the architecture is manually divided into four blocks based on its structural design. The overall network utilizes both the RGB modality $X_{img} \in \mathbb{R}^{3\times H\times W}$ and the high-frequency noise modality $X_{noise} \in \mathbb{R}^{3\times H\times W}$ processed by the MNFM as input. The MNFM comprises SRM [33], BayarConv [34], and NoisePrint++ [35], along with a fusion convolution, to aggregate features including local statistical anomalies, pixel-wise anomaly correlations, and device source noise.

Subsequently, the details of the Multi-stage Alternating Feature Extraction and Interaction are elaborated. At each stage, the dual-modal features $(R_i, N_i, i = 0, 1, 2, 3)$ from the previous level (where the initial input is regarded as level 0) are fed into the feature extraction blocks of the corresponding backbone networks, producing initial features for the next level $(R_{i+1}^{init}, N_{i+1}^{init})$:

$$R_{i+1}^{init} = GAE_{i+1}(R_i) \quad N_{i+1}^{init} = LAE_{i+1}(N_i) \tag{1}$$

The R_{i+1}^{init} are aligned with the noise modality. Both sets of features are then fed into the DFCM for deep interaction (as illustrated in Fig. 3):

$$(R_{i+1}, N_{i+1}) = DFCM_{i+1}(\delta(R_{i+1}^{init}), N_{i+1}^{init}) \tag{2}$$

where δ represents cross-modal Feature Alignment. The overall structure of the DFCM is symmetrically designed. First, the dual-modal features undergo lightweight channel attention for feature recalibration Γ. Channel-wise aggregation of the features is performed via adaptive average pooling, followed by the acquisition of channel weights through a projection head. The input features are then multiplied by these channel weights to obtain the recalibrated feature representation $(R_{i+1}^{mid}, N_{i+1}^{mid})$:

$$R_{i+1}^{mid} = \Gamma_{i+1}^{R}(\delta(R_{i+1}^{init})) \quad N_{i+1}^{mid} = \Gamma_{i+1}^{N}(N_{i+1}^{init}) \tag{3}$$

Subsequently, cross-attention is performed between the two modal features. Each modality serves as the query Q while the other provides the key-value pair (K, V), enabling bidirectional retrieval of intrinsic relationships and valuable information from the modalities. This mechanism enhances both the complementarity of the dual-modal features and their sensitivity to tampering artifacts. In contrast to classical cross-attention approaches, Group Convolution Modules (GCM) [36] are utilized as the transformation matrices for Q, K, and V, instead of applying simple linear transformations to the features prior to performing attention. This design not only reduces parameter and computational costs but also enables the capture of diverse feature patterns, similar to multi-head attention. It provides a unified representation space for features originating from different modalities, allowing cross-attention to compute inter-feature correlations more effectively. Finally, the output is generated by combining the result of the cross-attention with a residual connection of the original features, ensuring that the original information remains preserved:

$$\begin{aligned} R_{i+1} &= R_{i+1}^{init} + Attn_{R \leftarrow N}(G(R_{i+1}^{mid}), G(N_{i+1}^{mid}), G(N_{i+1}^{mid})) \\ N_{i+1} &= N_{i+1}^{init} + Attn_{N \leftarrow R}(G(N_{i+1}^{mid}), G(R_{i+1}^{mid}), G(R_{i+1}^{mid})) \end{aligned} \tag{4}$$

After being aligned back to the original feature space, the RGB modality features, together with the noise modality features, are returned to the corresponding backbone for the extraction of features at the next level.

3.3 Query-driven Multi-level Feature Decoding Mechanism

Existing approaches typically integrate upsampling with convolution or MLP. However, this strategy fails to pre-screen features, indiscriminately introducing substantial irrelevant background information and noise into the decoder. Consequently, subtle tampering features are severely overwhelmed. Inspired by the mechanism of the SAM Mask Decoder [25], we propose a lightweight, Query-driven Multi-level Feature Decoding. Specifically, a set of learnable tamper-aware queries $Q_{taq} \in \mathbb{R}^{N \times dim}$ is initialized at the beginning of the framework. These queries are subsequently utilized to generate corresponding query embeddings for each level of features. At each stage, the queries, together with the enhanced dual-modal features, are fed into the MFAM to be updated and generate the corresponding query embeddings $Q_i, i = 1, 2, 3, 4$:

$$Q_i = MFAM(Q_i^{init}, R_i, N_i) \tag{5}$$

where Q_i^{init} represents the initial state of the i-th level queries. The MFAM consists of three transformer blocks [37] (as shown in Fig. 3). The first two perform cross-attention, enabling the queries to establish richer and more robust cross-modal representations from both modalities while filtering out irrelevant features. The third block employs multi-head self-attention to further refine and integrate the information it has learned, thereby producing a final representation that is more semantically coherent and contextually enriched. Subsequently, the query embeddings Q_i are passed through an MLP layer to generate the queries Q_{i+1}^{init} for the next level, repeating the aforementioned process. The query embeddings at each level accumulate aggregated information from all preceding stages. This process is formulated as follows:

$$\begin{aligned} Q_i &= Trans_{Q \leftarrow N}(Q_i^{init}, R_i, N_i) \\ Q_i &= Trans_{Q \leftarrow R}(Q_i, N_i, R_i) \\ Q_i &= Trans_{Q \leftarrow Q}(Q_i, Q_i, Q_i) \\ Q_{i+1}^{init} &= MLP(Q_i) \end{aligned} \tag{6}$$

Then, the dual-modal features and their corresponding query embeddings from all levels are grouped into four triplets (Q_i, R_i, N_i) and fed into the QMFD. Its internal architecture is illustrated in the appendix. Within the decoder, the RGB modality features and noise modality features from the same level are fed into a Dilated Convolution Module (DCM) [38], leveraging its large receptive field to achieve preliminary fusion. The fused features $\{F_1^{fin} \sim F_4^{fin}\}$ are then multiplied with the corresponding query embeddings via matrix multiplication. This step is designed to perform feature selection. Finally, the features from all levels are upsampled to a common resolution and fed into the segmentation head ψ_{seg} to output the final localization mask $M_{loc} \in \mathbb{R}^{1 \times H \times W}$:

$$M_{loc} = \psi_{\text{seg}}(ConCat(Q_i \otimes F_i^{fin}, i = 1, 2, 3, 4)) \tag{7}$$

3.4 Loss Function Design

To enable the network to focus more on the edges of tampered areas, the RGB modality features output at each stage are fed into an Edge Supervision Module[26], which is a cascaded edge enhancement structure, to predict the boundaries of tampered regions, thereby constructing an edge loss $\mathcal{L}_{edg}$. Since the RGB features incorporate the LAE features obtained via DFCM, the edge loss can simultaneously supervise both branches. The predicted mask output from the decoder is used to construct the segmentation loss $\mathcal{L}_{seg}$. Considering the extreme imbalance between tampered and authentic pixels, Dice Loss [39] is employed to compute both $\mathcal{L}_{edg}$ and $\mathcal{L}_{seg}$. The total loss $\mathcal{L}_{total}$ of the model is formulated as follows, where, based on empirical practice, α is set to 0.2 and β to 0.8. G denotes the ground truth.

$$\mathcal{L}_{total} = \alpha\mathcal{L}_{seg}(M_{loc}, G_{loc}) + \beta\mathcal{L}_{edg}(M_{edg}, G_{edg}) \tag{8}$$

Table 1: Comparison of Pixel-level F1 and AUC across four datasets. Best results are **bold**, the second-best results are underlined.

Method	Pixel-level F1					Pixel-level AUC				
	CAS	COL	COV	NIST	AVG	CAS	COL	COV	NIST	AVG
ManTra-Net	0.327	0.462	0.196	0.193	0.295	0.643	0.724	0.566	0.709	0.661
PSCC-Net	0.578	0.822	0.416	0.341	0.539	0.918	0.919	0.872	0.810	0.880
MVSS-Net	0.583	0.723	0.470	0.320	0.524	0.904	0.911	0.868	0.777	0.865
CAT-Net	0.778	0.923	0.485	0.450	0.659	0.945	0.942	0.907	0.827	0.905
TruFor	0.700	0.903	0.379	0.426	0.602	0.951	0.936	0.887	0.863	0.909
IML-ViT	0.751	0.927	0.546	0.140	0.591	0.961	0.941	0.921	0.812	0.909
NFA-ViT	0.634	0.928	0.544	0.379	0.621	0.922	0.953	0.911	0.829	0.904
APSC-Net	0.798	0.941	0.402	**0.500**	0.660	0.916	0.942	0.731	0.775	0.841
SparseViT	0.819	**0.943**	0.521	0.390	0.668	0.952	**0.965**	0.910	0.844	0.917
Mesorch	0.829	0.911	0.530	0.408	0.670	0.969	0.914	0.907	**0.881**	0.918
MAMQ-Net (Ours)	**0.873**	0.939	**0.659**	0.480	**0.738**	**0.985**	0.943	**0.931**	0.860	**0.930**

4 Experiment

4.1 Experiment Setup

Training Dataset and Implementation Details Our model is trained on the standardized Protocol-CAT dataset [40], which consists of CASIAv2 [41], IMD2020[42], Fantastic Reality[43], and Tamp-COCO [44], totaling 826,556 images. These images cover multiple tampering types, such as splicing, copy-move, and removal. The detailed information for all datasets involved in this paper is provided in the appendix. Each image is resized to 512×512 for training input, and data augmentation methods such as random copy-move, flipping, and random JPEG compression are applied. The model is trained for 150 epochs with a batch size of 32 using two NVIDIA RTX 4090 GPUs. We adopt a cosine decay learning rate schedule, initialized at 1e-4 and gradually reduced to a minimum of 5e-7. The AdamW optimizer is employed with a weight decay of 0.05 to mitigate overfitting. All models are trained and evaluated on IMDL-Benco [45].

Test Dataset and Evaluation Metric The evaluation of our model is conducted on a series of public benchmarks, encompassing four widely adopted datasets: CASIAv1 [41], Coverage [46], NIST16 [47], Columbia [48]. These collections comprise images that exhibit a wide range of resolutions and incorporate diverse tampering strategies. In the appendix, we have added two additional AI-based editing datasets, AutoSplice[49] and COCOGlide[35], and evaluated our model on them. To quantitatively assess the model's performance in IFL, we employ the pixel-level F1 and AUC score as the primary evaluation metrics. All experimental results are obtained under a threshold of 0.5.

Figure 4: Robustness test results. The x-axis represents the attack intensity, while the y-axis denotes the pixel-level F1 score on the corresponding test datasets.

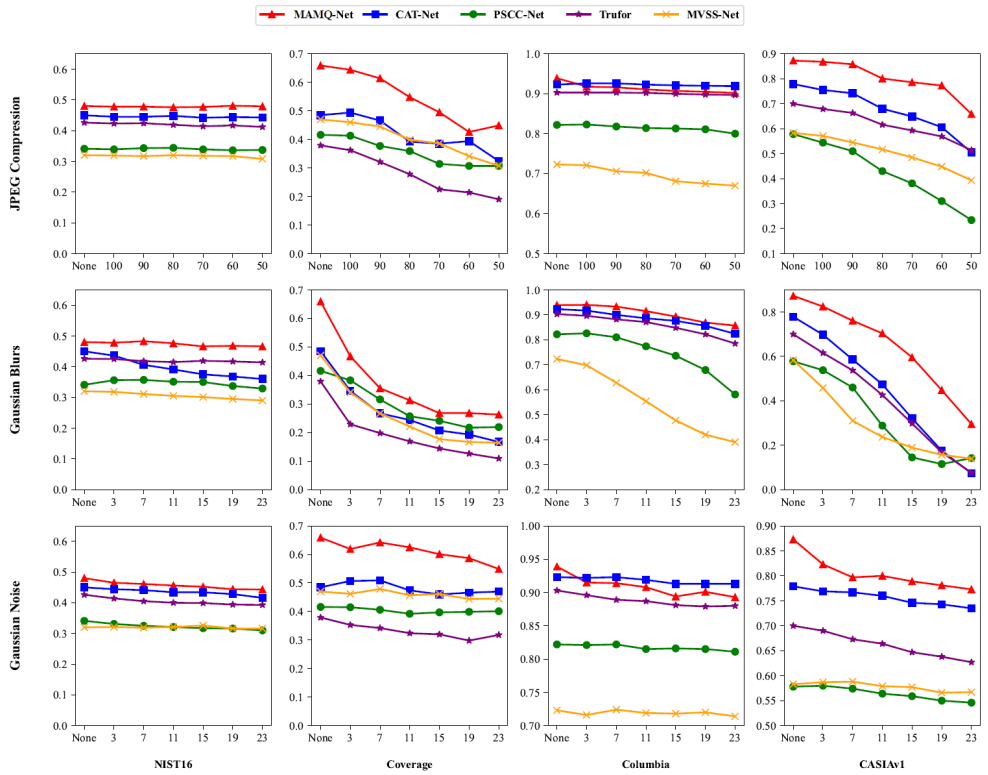


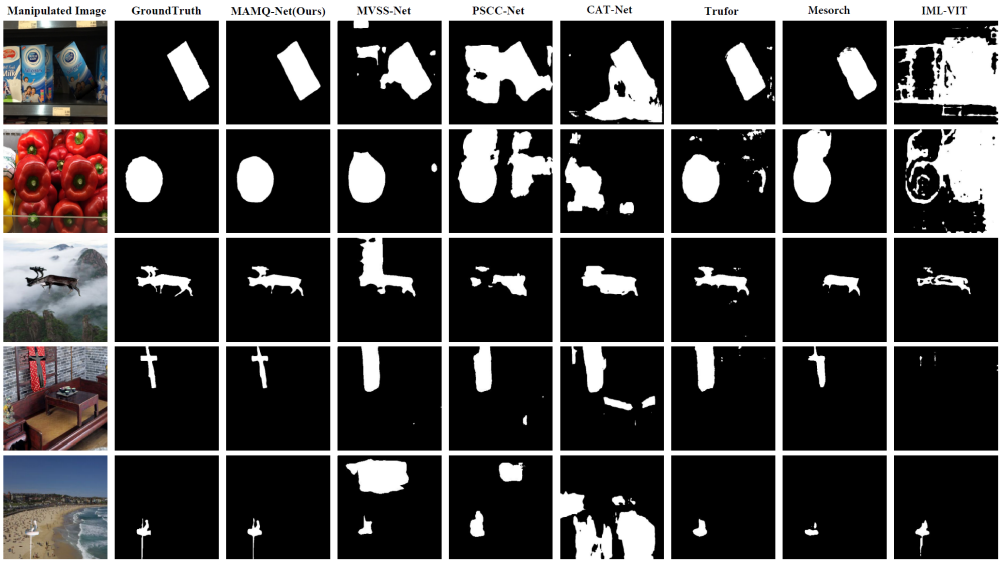
Table 2: Ablation Study Results. We conduct six additional sets of experiments to validate the effectiveness of the proposed components.

Method	**Setting**	**Pixel-level F1 Score**				
		CAS	**COL**	**COV**	**NIST**	**AVG**
Only backbone	RGB+Noise backbone	0.803	0.924	0.530	0.374	0.658
Ours w/ CP	w/o DFCM	0.850	0.943	0.579	0.456	0.707
Ours w/ SL	Only (Q_4, R_4, N_4)	0.849	0.943	0.604	0.416	0.703
Ours w/ SS	Only Stage#4	0.850	0.919	0.592	0.444	0.701
Ours w/ CD	Conv Decoder	0.857	0.946	0.582	0.438	0.706
Ours w/ MLP	MLP Decoder	0.859	0.936	0.601	0.445	0.710
MAMQ-Net	Ours	**0.873**	**0.939**	**0.659**	**0.480**	**0.738**

4.2 Performance Comparison with State-of-the-art

We adopt ManTra-Net [19], PSCC-Net [20], CAT-Net[44], MVSS-Net [26], TruFor [35], IML-ViT [5], NFA-ViT[50], APSC-Net[51], SparseViT[52] and Mesorch [1] as baseline methods for comparison. For fairness, all baseline models are retrained on the Protocol-CAT dataset. The corresponding experimental results are reported in Table 1. As shown in the table, MAMQ-Net achieves optimal or near-optimal results on all four datasets and delivers the best average performance. In addition, Fig. 5 illustrates a qualitative comparison of the predicted results from our model and the competing approaches. It can be clearly observed that our framework delineates the boundaries of manipulated regions more accurately, leading to fewer false alarms and higher precision. This fact demonstrates that our framework successfully constructs complementary dual-modal representations sensitive to tampering and accurately localizes the tampered regions.

Figure 5: IFL results on multiple datasets. The leftmost two columns are the manipulated image and groundtruth, followed by the prediction results of different models.



4.3 Robustness Study

To assess the robustness of the model against different attack scenarios, we apply degradation operations—including Gaussian noise, Gaussian blur, and JPEG compression—to the tampered images. The corresponding results are depicted in Fig. 4. Our framework consistently surpasses other SoTA approaches on the CASIAv1, Coverage, and NIST16 datasets across all attack types. For the Columbia, our method achieves superior performance under Gaussian blur and is only marginally outperformed by CAT-Net under the other two degradations. These findings collectively demonstrate the strong robustness of our model. Further quantitative results of the robustness test are provided in the appendix.

4.4 Ablation Study

Ablation study on RGB and Noise backbone To demonstrate the synergistic effect of our proposed components, we compare the performance of the backbone-only model and the full MAMQ-Net, with the results presented in Table 2. Compared with the backbone-only baseline (Setting 1), the full MAMQ-Net improves Avg.F1 from 0.658 to 0.738 (+8.0 points) and yields consistent gains on all four datasets, with the most pronounced improvements on COV (+12.9) and NIST (+10.6). This confirms the effectiveness of the proposed components, especially on the more challenging benchmarks.

Ablation study on MNFM In the MNFM, we posit that a simple 3x3 convolution is sufficient to fuse the information derived from different noise extractors. We conduct three additional sets of experiments (using only SRM, Bayar, or NoisePrint++) to verify whether the model truly exploits complementary rather than conflicting noise information. The results are presented in Table 3. Using them jointly yields better results on average than using them individually, demonstrating that we have, to some extent, resolved the potential conflicts among different noise extractors.

Table 3: The results of ablation study on MNFM. Best results are **bold**.

MNFM	Pixel-level F1 Score				
	CAS	COL	COV	NIST	AVG
SRM	**0.874**	**0.960**	0.620	0.449	0.726
Bayar	0.862	0.944	0.580	0.443	0.707
NP++	0.844	0.922	0.633	0.446	0.711
SRM+Bayar+NP++ (Ours)	0.873	0.939	**0.659**	**0.480**	**0.738**

Ablation study on Multi-stage Alternating Feature Extraction and Interaction We validate the effectiveness of the proposed Multi-stage Alternating Feature Extraction and Interaction paradigm from two perspectives. On one hand, we remove DFCM at each stage; on the other hand, we only retain the fourth stage. The results are listed in Settings 2 and 4 of Table 2 respectively. Our

model demonstrates an average performance improvement of 4.4% and 5.3% compared to these two scenarios, respectively. This fact indicates that our method effectively constructs complementary bimodal tampering representations, enhances sensitivity to tampering artifacts, and suppresses intra-modal noise.

Ablation study on Query-driven Multi-level Feature Decoding We replace the Query-driven Multi-level Feature Decoding with upsampling followed by convolutional or MLP decoders. The results are shown in Settings 5 and 6 in Table 2. Our model shows improvements of 4.5% and 3.9% compared to these two scenarios, respectively. Furthermore, compared to using single-level features (setting 3), our model achieves a 5.0% performance improvement. These findings indicate the necessity of each level of features and that our method achieves effective feature selection and aggregation. Additionally, we vary the number of queries to 4, 8, and 32 to study the impact of query quantity on model performance(Table 4). With 16 queries, the average p-F1 improves by 2.8%, 2.4%, and 3.1% over the 4-, 8-, and 32-query configurations, respectively. It is noted that performance is non-monotonic with query quantity, following a unimodal curve. A detailed analysis of the possible reasons is provided in the Appendix.

Table 4: The results of ablation study on query quantity. Best results are **bold**, second-best are underlined.

Queries	Pixel-level F1 Score				
	CAS	COL	COV	NIST	AVG
4 queries	0.864	0.942	0.613	0.450	0.717
8 queries	0.857	0.933	0.641	0.448	0.720
32 queries	0.861	**0.952**	0.590	0.457	0.715
16 queries (Ours)	**0.873**	0.939	**0.659**	**0.480**	**0.738**

5 Model Complexity Analysis

We report the parameter counts, FLOPs, latency, and performance of all baselines and our model in the Table 5. All results are measured with a batch size of 1 on identical hardware (CPU, GPU, etc.). MAMQ-Net† denotes a lightweight variant that removes the relatively time-consuming NoisePrint++ from the full version. MAMQ-Net achieves the best overall performance with an Avg.F1 of 0.738, outperforming the strongest baselines Mesorch (0.670) and SparseViT (0.668) by about 7 points, while keeping parameters and FLOPs comparable to other high-accuracy methods. Its lightweight variant, MAMQ-Net†, further cuts FLOPs by 63.4% and latency by 31.9% with only a 0.010 F1 drop, yielding a more favorable accuracy-efficiency trade-off than lightweight competitors such as Mesorch and SparseViT at similar latency.

6 Conclusion

In this work, we propose a novel IFL network, MAMQ-Net, which consists of a Multi-stage Alternating Feature Extraction and Interaction architecture and a lightweight, Query-driven Multi-level Feature Decoding. The former simulates the cognitive processes of the human brain, constructing complementary tampering feature representations through cross-modal Feature Alignment and DFCM at each stage, thereby enhancing sensitivity to tampering artifacts. Moreover, the latter employs learnable tamper-aware queries to progressively aggregate crucial information from multi-level features through MFAM at each stage. In QMFD, these query embeddings perform selective refinement and aggregation of multi-level features to accurately predict tampered regions. Extensive experiments demonstrate that our framework outperforms current SoTA models in localization accuracy and robustness across multiple public datasets, achieving an optimal balance between performance and complexity.

Table 5: Comparison of parameters and computational efficiency (FLOPs) across different models.

Method	Parameters (M)	FLOPs (G)	Latency (ms)	Avg.F1
ManTra-Net	3.9	274.0	19.96	0.295
MVSS-Net	150.5	171.0	29.64	0.524
PSCC-Net	3.7	376.8	11.32	0.539
CAT-Net	116.7	137.2	42.03	0.659
TruFor	68.7	236.5	34.98	0.602
IML-ViT	91.8	577.0	66.43	0.591
APSC-Net	139.8	416.0	25.72	0.660
SparseViT	50.2	41.5	16.05	0.668
Mesorch	85.8	124.9	20.96	0.670
NFA-ViT	60.3	156.7	30.13	0.621
MAMQ-Net	**114.0**	**230.0**	**31.29**	**0.738**
MAMQ-Net†	**112.0**	**84.3**	**21.31**	**0.728**

7 Acknowledgements and Disclosures

This work is supported by "National Key Research and Development Program of China" [2024YFF0907404]. The authors declare no competing interests.

References

[1] X. Zhu, X. Ma, L. Su, Z. Jiang, B. Du, X. Wang, Z. Lei, W. Feng, C.-M. Pun, and J.-Z. Zhou, "Mesoscopic insights: orchestrating multi-scale & hybrid architecture for image manipulation localization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 10, 2025, pp. 11 022–11 030.

[2] N. Zeng, P. Wu, Y. Zhang, H. Li, J. Mao, and Z. Wang, "Dpmsn: a dual-pathway multiscale network for image forgery detection," *IEEE Transactions on Industrial Informatics*, vol. 20, no. 5, pp. 7665–7674, 2024.

[3] K. Guo, H. Zhu, and G. Cao, "Effective image tampering localization via enhanced transformer and co-attention fusion," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2024, pp. 4895–4899.

[4] X. Lin, S. Wang, J. Deng, Y. Fu, X. Bai, X. Chen, X. Qu, and W. Tang, "Image manipulation detection by multiple tampering traces and edge artifact enhancement," *Pattern Recognition*, vol. 133, no. 1, p. 109026, 2023.

[5] X. Ma, B. Du, Z. Jiang, A. Y. A. Hammadi, and J. Zhou, "Iml-vit: benchmarking image manipulation localization by vision transformer," *arXiv preprint arXiv:2307.14863*, 2023.

[6] P. Zhuang, H. Li, S. Tan, B. Li, and J. Huang, "Image tampering localization using a dense fully convolutional network," *IEEE Transactions on Information Forensics and Security*, vol. 16, no. 4, pp. 2986–2999, 2021.

[7] L. Zhang, X. Zhu, D. He, X. Liao, and B. Sun, "Samif: adapting segment anything model for image inpainting forensics," in *Proceedings of the Asian Conference on Computer Vision*, 2024, pp. 3605–3621.

[8] Y. Su, S. Tan, and J. Huang, "A novel universal image forensics localization model based on image noise and segment anything model," in *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security*, 2024, pp. 149–158.

[9] M.-J. Kwon, W. Lee, S.-H. Nam, M. Son, and C. Kim, "Safire: segment any forged image region," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 4, 2025, pp. 4437–4445.

[10] H. Zhang, Q. He, X. Bi, W. Li, B. Liu, and B. Xiao, "Towards universal ai-generated image detection by variational information bottleneck network," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 23 828–23 837.

[11] P. Ferrara, T. Bianchi, A. De Rosa, and A. Piva, "Image forgery localization via fine-grained analysis of cfa artifacts," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 5, pp. 1566–1577, 2012.

[12] S. Ye, Q. Sun, and E.-C. Chang, "Detecting digital image forgeries by measuring inconsistencies of blocking artifact," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, 2007, pp. 12–15.

[13] D. Tralic, J. Petrovic, and S. Grgic, "Jpeg image tampering detection using blocking artifacts," in *Proceedings of the International Conference on Systems, Signals and Image Processing*, 2012, pp. 5–8.

[14] C. Kong, A. Luo, S. Wang, H. Li, A. Rocha, and A. C. Kot, "Pixel-inconsistency modeling for image manipulation localization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 47, no. 6, pp. 4455–4472, 2025.

[15] D. Cozzolino and L. Verdoliva, "Noiseprint: a cnn-based camera model fingerprint," *IEEE Transactions on Information Forensics and Security*, vol. 15, no. 5, pp. 144–159, 2020.

[16] J. H. Bappy, C. Simons, L. Nataraj, B. S. Manjunath, and A. K. Roy-Chowdhury, "Hybrid lstm and encoder–decoder architecture for detection of image forgeries," *IEEE Transactions on Image Processing*, vol. 28, no. 7, pp. 3286–3300, 2019.

[17] J. Hao, Z. Zhang, S. Yang, D. Xie, and S. Pu, "Transforensics: Image forgery localization with dense self-attention," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15 035–15 044.

[18] X. Hu, Z. Zhang, Z. Jiang, S. Chaudhuri, Z. Yang, and R. Nevatia, "Span: spatial pyramid attention network for image manipulation localization," in *Proceedings of the European Conference on Computer Vision*, 2020, pp. 312–328.

[19] Y. Wu, W. AbdAlmageed, and P. Natarajan, "Mantra-net: manipulation tracing network for detection and localization of image forgeries with anomalous features," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9535–9544.

[20] X. Liu, Y. Liu, J. Chen, and X. Liu, "Pscc-net: progressive spatio-channel correlation network for image manipulation detection and localization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 11, pp. 7505–7517, 2022.

[21] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, W. Liu, and B. Xiao, "Deep high-resolution representation learning for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 10, pp. 3349–3364, 2021.

[22] J. Wang, Z. Wu, J. Chen, X. Han, A. Shrivastava, S.-N. Lim, and Y.-G. Jiang, "Objectformer for image manipulation detection and localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2354–2363.

[23] Z. Xu, X. Zhang, R. Li, Z. Tang, Q. Huang, and J. Zhang, "Fakeshield: explainable image forgery detection and localization via multi-modal large language models," in *Proceedings of the International Conference on Learning Representations*, 2025.

[24] Q. Zhang, Y. Qi, X. Tang, J. Fang, X. Lin, K. Zhang, and C. Yuan, "IMDPrompter: adapting SAM to image manipulation detection by cross-view automated prompt learning," in *Proceedings of the International Conference on Learning Representations*, 2025.

[25] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 3992–4003.

[26] C. Dong, X. Chen, R. Hu, J. Cao, and X. Li, "Mvss-net: multi-view multi-scale supervised networks for image manipulation detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 3, pp. 3539–3553, 2023.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[28] A. Mazumdar and P. K. Bora, "Two-stream encoder–decoder network for localizing image forgeries," *Journal of Visual Communication and Image Representation*, vol. 82, no. 1, p. 103417, 2022.

[29] Y. Niu, P. Chen, L. Zhang, L. Tan, and Y. Chen, "Image forgery localization via guided noise and multi-scale feature aggregation," *arXiv preprint arXiv:2412.01622*, 2024.

[30] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 1280–1289.

[31] Y. Liu, C. Liu, K. Han, Q. Tang, and Z. Qin, "Boosting semantic segmentation from the perspective of explicit class embeddings," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 821–831.

[32] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "Segformer: simple and efficient design for semantic segmentation with transformers," *Advances in Neural Information Processing Systems*, vol. 34, no. 11, pp. 12 077–12 090, 2021.

[33] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis, "Learning rich features for image manipulation detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1053–1061.

[34] B. Bayar and M. C. Stamm, "Constrained convolutional neural networks: a new approach towards general purpose image manipulation detection," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2691–2706, 2018.

[35] F. Guillaro, D. Cozzolino, A. Sud, N. Dufour, and L. Verdoliva, "Trufor: leveraging all-round clues for trustworthy image forgery detection and localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 20 606–20 615.

[36] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, no. 5, pp. 84–90, 2012.

[37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, no. 12, pp. 6000–6010, 2017.

[38] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, and G. Cottrell, "Understanding convolution for semantic segmentation," in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, 2018, pp. 1451–1460.

[39] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: fully convolutional neural networks for volumetric medical image segmentation," in *Proceedings of the International Conference on 3D Vision*, 2016, pp. 565–571.

[40] M.-J. Kwon, I.-J. Yu, S.-H. Nam, and H.-K. Lee, "Cat-net: compression artifact tracing network for detection and localization of image splicing," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 375–384.

[41] J. Dong, W. Wang, and T. Tan, "Casia image tampering detection evaluation database," in *Proceedings of the IEEE China Summit and International Conference on Signal and Information Processing*, 2013, pp. 422–426.

[42] A. Novozamsky, B. Mahdian, and S. Saic, "Imd2020: a large-scale annotated dataset tailored for detecting manipulated images," in *Proceedings of the IEEE Winter Applications of Computer Vision Workshops*, 2020, pp. 71–80.

[43] V. V. Kniaz, V. Knyaz, and F. Remondino, "The point where reality meets fantasy: Mixed adversarial generators for image splice detection," *Advances in Neural Information Processing Systems*, vol. 32, no. 12, pp. 215–226, 2019.

[44] M.-J. Kwon, S.-H. Nam, I.-J. Yu, H.-K. Lee, and C. Kim, "Learning jpeg compression artifacts for image manipulation detection and localization," *International Journal of Computer Vision*, vol. 130, no. 8, pp. 1875–1895, 2022.

[45] X. Ma, X. Zhu, L. Su, B. Du, Z. Jiang, B. Tong, Z. Lei, X. Yang, C.-M. Pun, J. Lv *et al.*, "Imdl-benco: a comprehensive benchmark and codebase for image manipulation detection & localization," *Advances in Neural Information Processing Systems*, vol. 37, no. 6, pp. 134 591–134 613, 2024.

[46] B. Wen, Y. Zhu, R. Subramanian, T.-T. Ng, X. Shen, and S. Winkler, "Coverage—a novel database for copy-move forgery detection," in *Proceedings of the IEEE International Conference on Image Processing*, 2016, pp. 161–165.

[47] H. Guan, M. Kozak, E. Robertson, Y. Lee, A. N. Yates, A. Delgado, D. Zhou, T. Kheyrkhah, J. Smith, and J. Fiscus, "Mfc datasets: large-scale benchmark datasets for media forensic challenge evaluation," in *Proceedings of the IEEE Winter Applications of Computer Vision Workshops*, 2019, pp. 63–72.

[48] Y.-F. Hsu and S.-F. Chang, "Detecting image splicing using geometry invariants and camera characteristics consistency," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, 2006, pp. 549–552.

[49] S. Jia, M. Huang, Z. Zhou, Y. Ju, J. Cai, and S. Lyu, "Autosplice: A text-prompt manipulated image dataset for media forensics," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 893–903.

[50] L. Cai, H. Wang, J. Ji, Y. ZhouMen, Y. Ma, X. Sun, L. Cao, and R. Ji, "Zooming in on fakes: A novel dataset for localized ai-generated image detection with forgery amplification approach," *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2026.

[51] C. Qu, Y. Zhong, C. Liu, G. Xu, D. Peng, F. Guo, and L. Jin, "Towards modern image manipulation localization: A large-scale dataset and novel methods," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 10 781–10 790.

[52] L. Su, X. Ma, X. Zhu, C. Niu, Z. Lei, and J.-Z. Zhou, "Can we get rid of handcrafted feature extractors? sparsevit: Nonsemantics-centered, parameter-efficient image manipulation localization through spare-coding transformer," *arXiv preprint arXiv:2412.14598*, 2024.

[53] H. Zhao, Y. Zhuge, Y. Wang, L. Wang, H. Lu, and Y. Zeng, "Learning universal features for generalizable image forgery localization," *arXiv preprint arXiv:2504.07462*, 2025.

A Detailed information about the datasets used in this paper

We trained our model using the protocol-CAT dataset, which was first introduced and utilized by CAT-Net. It consists of four datasets: CASIAv2, IMD2020, TampCOCO and Fantastic Reality. CASIAv2 provides high-quality copy-move and spliced tampered images. IMD2020 includes complex real-world edits (such as splicing and local modifications) with non-fixed resolutions. Fantastic Reality is a multi-task annotation dataset that integrates tamper localization and semantic segmentation, providing pixel-level tamper region masks, instance segmentation, and category labels. TampCOCO is constructed based on the COCO 2017 dataset and includes two parts: copy-move and splicing. All images undergo JPEG compression while retaining clear boundaries to support model learning of low-level tampering traces. The information of all training sets is listed in Table 6. These training sets contain rich semantic details and noise patterns, making them more aligned with real-world application scenarios. By training on these datasets, we can learn more comprehensive and hierarchical features of tampering, effectively enhancing both robustness and generalization capabilities.

We select six public benchmark datasets as our test data, namely CASIAv1, Columbia, Coverage, NIST16, COCOGlide, and AutoSplice. CASIAv1 primarily provides high-quality spliced images. The Columbia dataset focuses on uncompressed spliced images and features high resolution. The Coverage dataset addresses copy-move forgeries, typically by copying and pasting one item from a group of similar objects within an image.The NIST16 comprises three forgery types: splicing, removal, and copy-move operations, and maintains high image resolution throughout. The COCOGlide focuses on generative image forgery research by combining the GLIDE diffusion model with semantic prompts to create tampered content, simulating semantic-level local manipulations. The AutoSplice represents a text-prompt manipulated image collection where all images undergo JPEG compression processing. The information of all test sets is listed in Table 7. These datasets encompass diverse manipulation types, exhibit wide resolution ranges, and contain varied forgery region sizes, collectively enabling comprehensive evaluation of model performance across multiple dimensions.

B The details of QMFD

The internal architecture of QMFD is illustrated in Fig. 6. The QMFD comprises four branches, each dedicated to processing dual-modal features at four distinct levels. Given a ternary tuple (Q_i, R_i, N_i)

Table 6: Detailed information of the protocol-CAT training set, where "N/K" indicates that the quantity of this type is unknown.

Dataset	Manipulation type			Number	Resolution	
	copy-move	splice	remove		min	max
CASIAv2	3274	1849	0	5123	320×240	800×600
IMD2020	N/K	N/K	N/K	2010	260×193	2958×4437
TampCOCO	600000	200000	0	800000	72×51	640×640
Fantastic Reality	N/K	N/K	N/K	19423	500×333	6000×4000
Total	N/K	N/K	N/K	826556	72×51	6000×4000

Table 7: Detailed information of the six public benchmark datasets, where "N/K" indicates that the quantity of this type is unknown.

Dataset	Manipulation type				Number	Resolution	
	copy-move	splice	remove	AI-Gen		min	max
CASIAv1	459	461	0	0	920	384×256	384×256
Columbia	0	180	0	0	180	757×568	1152×768
Coverage	100	0	0	0	100	334×190	752×472
NIST16	236	225	103	0	564	500×500	5616×3744
COCOGlide	0	0	0	512	512	256×256	256×256
AutoSplice	0	0	0	3621	3621	256×256	4232×4232
Total	795	866	103	4133	5897	334×190	5616×3744

at one level, the dual-modal features are fused through the DCM to produce the fused features F_i^{fin}. The query Q_i then performs pixel-wise feature weighting and reconstruction across the F_i^{fin}. Specifically, each query shares the same length as the pixel's feature vector, enabling element-wise weighted summation operations. This step aims to emphasize critical information while suppressing irrelevant feature interference. Finally, the refined features from all four levels are fed into a prediction head to output the tampered region mask. The prediction head consists of a simple 1×1 convolutional layer.

C Comparison with baseline models using additional metric

We conduct supplementary experiments on four benchmark datasets using permute-F1 and pixel-level IoU scores as evaluation metrics. The results are presented in Table 8. As shown in the results, MAMQ-Net achieves SoTA performance across all metrics, with particularly significant improvements in average permute-F1 and pixel-level IoU scores compared to other models. This outcome further validates that our method successfully constructs complementary dual-modal features sensitive to tampering artifacts, and effectively filters and aggregates multi-level features through the query-driven mechanism.

D The model's performance on AI-generated tampering techniques

We supplement the performance of MAMQ-Net on COCOGlide and AutoSplice (image tampering based on diffusion models or LLM) to evaluate localization capabilities for AI-based tampered images. As shown in the Table 9, MAMQ-Net achieves optimal or sub-optimal performance across all metrics on both datasets.

E The model's performance on GAN-based tampering techniques

We train and evaluate our model on the ForgeryADE[53] dataset, which contains images tampered with by four mainstream GAN models, including Crfill, Ctsdg, Deepfillv2 and LaMa. The test results

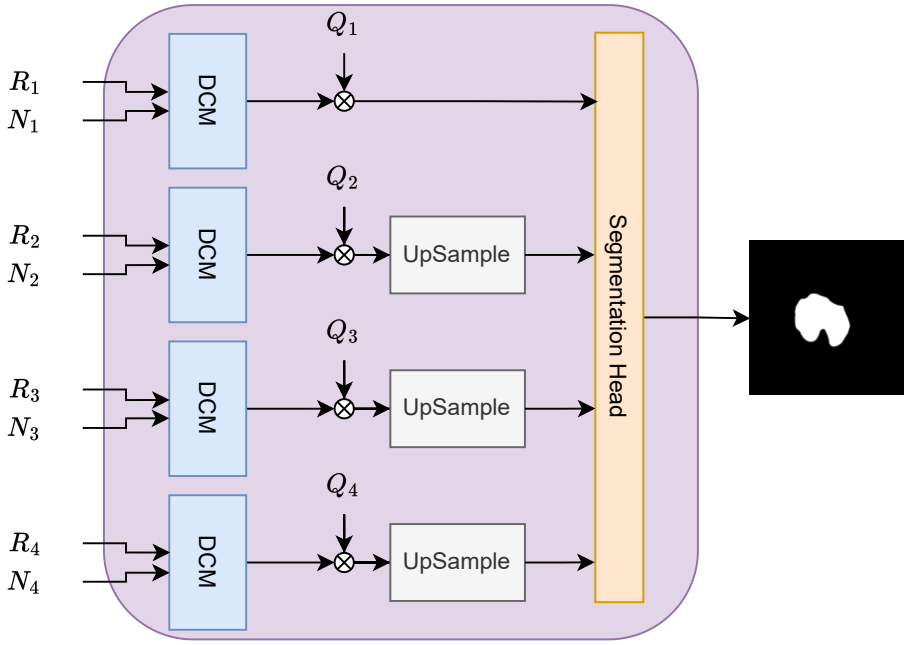


Figure 6: The pipeline of QMFD. A DCM and multi-level query embeddings are leveraged to pre-screen features, followed by concatenation of features from all levels into the prediction head for precise localization of tampered regions.

Table 8: Comparison of permute-F1 and Pixel-level IoU across four datasets. Best results are **bold**, second-best are underlined.

Method	Permute-F1					Pixel-level IoU				
	CAS	COL	COV	NIST	AVG	CAS	COL	COV	NIST	AVG
PSCC-Net	0.589	0.830	0.451	0.371	0.560	0.442	0.729	0.307	0.259	0.434
MVSS-Net	0.597	0.768	0.529	0.357	0.563	0.481	0.641	0.397	0.236	0.439
CAT-Net	0.786	0.951	0.551	0.490	0.695	0.716	0.903	0.430	0.388	0.609
TruFor	0.714	0.934	0.443	0.466	0.639	0.621	0.874	0.303	0.350	0.537
IML-ViT	0.763	0.974	0.595	0.169	0.625	0.683	0.919	0.482	0.127	0.553
Mesorch	0.837	0.966	0.594	0.467	0.716	0.778	0.896	0.481	0.353	0.627
MAMQ-Net (Ours)	**0.880**	**0.995**	**0.697**	**0.520**	**0.773**	**0.832**	0.935	**0.617**	**0.420**	**0.701**

are listed in Table 10. MAMQ-Net achieves the best pixel-level F1 on all four GAN-based tampering methods, with an Avg.F1 of 0.873, about 3 points higher than the second-best Mesorch (0.840) and CAT-Net (0.839). The gain is most pronounced on LaMa (0.883 vs. 0.837, +4.6 points), where IML-ViT and TruFor drop noticeably to 0.744 and 0.675, respectively, indicating that MAMQ-Net generalizes more stably across different GAN-based traces.

F Quantitative results of robustness study

We quantify the robustness test data presented in Fig. 4, using the average pixel-level F1 score under single-attack types with varying intensity factors as the evaluation metric. The results are presented in Tables 11 and 12. We apply degradation techniques such as Gaussian noise (GN) with different standard deviations(3,7,11,15,19,23), Gaussian blur (GB) with varying kernel sizes (3,7,11,15,19,23), and JPEG compression (JC) with different quality factors (100,90,80,70,60,50) to the tampered

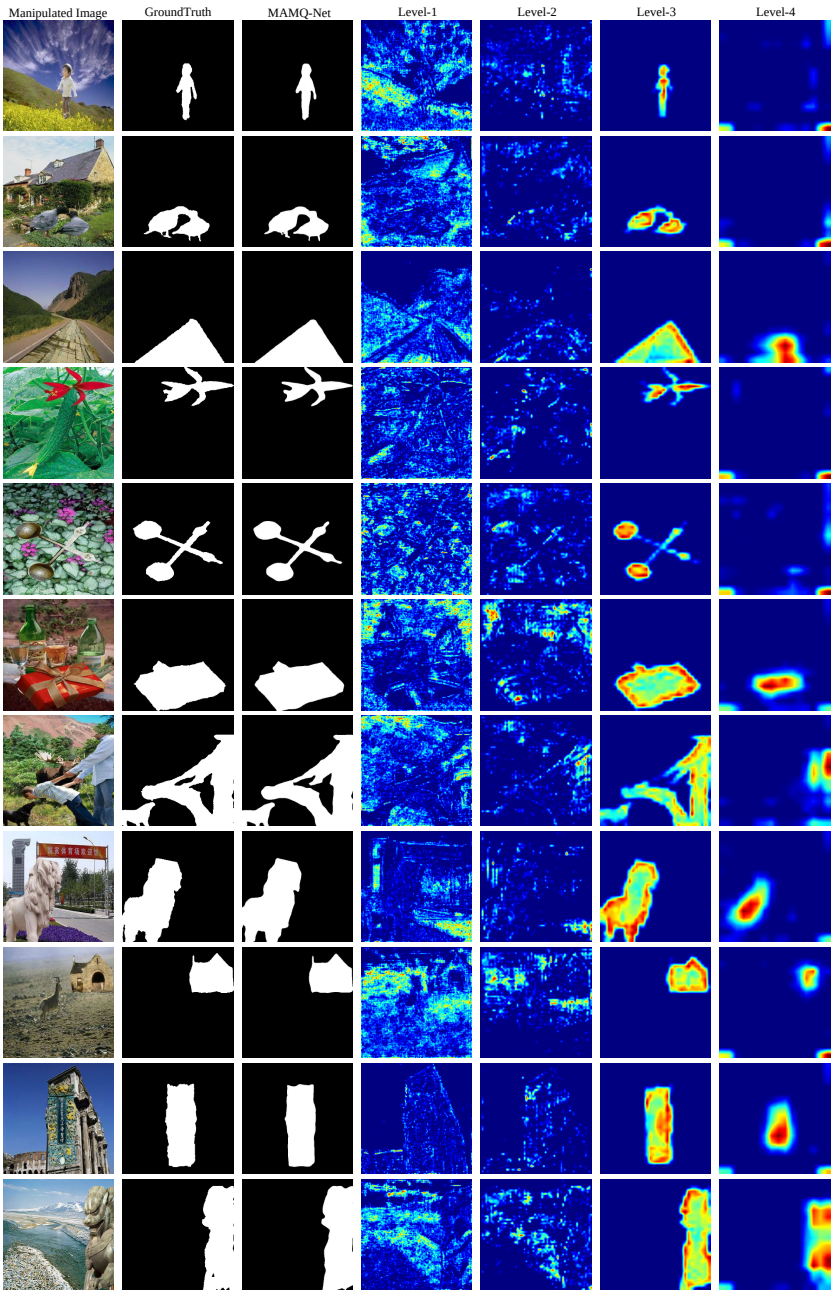


Figure 7: Visualization of Grad-CAM for multi-level features. The first three columns are the tampered image, groundtruth, and prediction results, respectively. Followed by the corresponding fused multi-level features.

images. As visually demonstrated in both tables, MAMQ-Net achieves SoTA performance on CASIAv1, Coverage, and NIST16 datasets across all attack types.

G Robustness to Scale Perturbations

We test the robustness of different models under scale perturbations with various scaling factors (0.5,0.75,1,1.25,1.5), and the results on the CASIAv1 and Columbia dataset are shown in Table 13. MAMQ-Net consistently and significantly outperforms the compared models across most

Table 9: Comparison on COCOGlide and AutoSplice with different SoTA models. Best results are **bold**, second-best are underlined.

Method	COCOGlide			AutoSplice		
	F1	AUC	IoU	F1	AUC	IoU
MVSS-Net	0.428	0.819	0.327	0.388	0.755	0.272
PSCC-Net	0.458	0.848	0.396	**0.455**	0.871	**0.406**
CAT-Net	0.409	0.849	0.334	0.450	0.862	0.348
IML-ViT	0.369	0.835	0.290	0.343	0.854	0.246
Mesorch	0.397	**0.894**	0.329	0.357	**0.926**	0.252
MAMQ-Net (Ours)	**0.477**	0.867	**0.416**	0.451	0.875	0.352

Table 10: Pixel-level F1 scores on images manipulated by different inpainting methods.

Model	Crfill	Ctsdg	Deepfillv2	LaMa	Avg.F1
CAT-Net	0.805	0.864	0.850	0.837	0.839
IML-ViT	0.796	0.845	0.840	0.744	0.806
Mesorch	0.814	0.854	0.857	0.835	0.840
TruFor	0.642	0.721	0.729	0.675	0.692
MAMQ-Net (Ours)	**0.848**	**0.883**	**0.878**	**0.883**	**0.873**

Table 11: Average pixel-level F1 comparison on CASIAv1 and Columbia datasets under different attacks. Best results are **bold**, second-best are underlined.

Method	CASIAv1			Columbia		
	JC	GB	GN	JC	GB	GN
PSCC-Net	0.427	0.324	0.564	0.814	0.747	0.817
MVSS-Net	0.506	0.296	0.578	0.697	0.556	0.719
CAT-Net	0.674	0.444	0.757	**0.923**	0.883	**0.918**
TruFor	0.619	0.403	0.663	0.901	0.858	0.888
MAMQ-Net (Ours)	**0.803**	**0.643**	**0.805**	0.914	**0.907**	0.909

Table 12: Average pixel-level F1 comparison on NIST16 and Coverage datasets under different attacks. Best results are **bold**, second-best are underlined.

Method	NIST16			Coverage		
	JC	GB	GN	JC	GB	GN
PSCC-Net	0.340	0.346	0.323	0.356	0.293	0.404
MVSS-Net	0.317	0.306	0.320	0.401	0.258	0.460
CAT-Net	0.445	0.398	0.435	0.420	0.273	0.482
TruFor	0.419	0.419	0.404	0.281	0.193	0.333
MAMQ-Net (Ours)	**0.478**	**0.474**	**0.457**	**0.548**	**0.370**	**0.612**

individual perturbation factors and in overall performance, demonstrating strong robustness to scale perturbations.

Table 13: F1 scores under different scale perturbation factors on CASIAv1 and Columbia.

Method	CASIAv1						Columbia					
	0.5	0.75	1	1.25	1.5	Avg.F1	0.5	0.75	1	1.25	1.5	Avg.F1
MVSS-Net	0.151	0.223	0.583	0.489	0.385	0.366	0.445	0.712	0.723	0.612	0.528	0.604
PSCC-Net	0.131	0.150	0.578	0.485	0.429	0.355	0.369	0.429	0.822	0.767	0.705	0.618
CAT-Net	0.154	0.458	0.778	0.694	0.600	0.537	0.400	0.493	0.923	0.885	0.787	0.698
TruFor	0.329	0.371	0.700	0.644	0.575	0.524	0.471	0.593	0.903	0.873	0.770	0.722
MAMQ-Net (Ours)	**0.421**	**0.745**	**0.873**	**0.814**	**0.722**	**0.715**	**0.531**	**0.892**	**0.939**	**0.894**	**0.822**	**0.816**

H Grad-CAM analysis of multi-level tamper-aware queries

We visualized the multi-level features modulated by tamper-aware queries using Grad-CAM in Fig. 7. Red areas represent the high-response regions in the feature maps. It can be observed that Level-1 and Level-2 tamper-aware queries focus on the edges of the tampered regions, effectively capturing low-level features. In contrast, the deeper Level-3 and Level-4 tamper-aware queries focus on high-level object-based characteristics without interference from other regions. This phenomenon demonstrates that our model successfully constructs complementary dual-modal features sensitive to tampering artifacts, and effectively aggregates these dual-modal features through query-driven mechanisms, thereby suppressing interference from irrelevant features.

I Ablation study on DFCM

We replace the DFCM with concatenation and element-wise addition and retrain the model. The results are shown in Table 14. It can be observed that, compared to these simpler methods, the DFCM does not introduce significant additional parameters or computational complexity, yet achieves the best performance. This advantage is attributed to our deliberate design.

Table 14: The results of ablation study on DFCM. Best results are **bold**.

Interaction method	**Parameters(M)**	**FLOPS (G)**	**Pixel-level F1 Score**				
			CAS	**COL**	**COV**	**NIST**	**AVG**
Element-wise addition	**105.0**	**227.0**	0.867	0.925	0.616	0.468	0.719
Concatenation	112.0	232.0	0.862	**0.951**	0.628	0.448	0.722
DFCM (Ours)	114.0	230.0	**0.873**	0.939	**0.659**	**0.480**	**0.738**

J Ablation study on query quantity

It is noted that the model's performance does not monotonically improve with increasing query quantity, but rather follows a unimodal curve pattern. The non-semantic features of an image, such as frequency and noise, are highly sensitive to manipulation and show greater independence across different regions of the image. We hypothesize that the underlying reason is that an excessive number of queries hinders the model's ability to effectively capture these non-semantic features. The limited tampering feature signals are forced to disperse across a larger number of query vectors. As a result, each query may only capture subtler, more fragmented cues, failing to form a comprehensive and highly discriminative representation of the tampered regions.

K Ablation study on interaction order in DFCM

We retrained the model by changing the interaction order to "Noise-aligned-to-RGB" in DFCM, and the results (pixel-F1) in Table 15 show that the "RGB-aligned-to-Noise" interaction order yields

better performance. We hypothesize that this is because noise features are compact, whereas RGB features contain a wealth of detailed information. Attempting to align noise features to match RGB features may introduce a significant amount of irrelevant or misleading information.

Table 15: The results of ablation study on interaction order in DFCM. Best results are **bold**.

Interaction order	**Pixel-level F1 Score**				
	CAS	**COL**	**COV**	**NIST**	**AVG**
Noise-aligned-to-RGB	0.867	0.922	0.596	0.475	0.715
RGB-aligned-to-Noise (Ours)	**0.873**	0.939	**0.659**	**0.480**	**0.738**

L Ablation Study on Loss Function Weights

Since a large number of tampering artifacts exist at the edges of tampered regions, we assign a larger weight to the edge loss, expecting the model to focus on the edges of tampered regions. Meanwhile, we also conduct additional ablation experiments to further demonstrate the rationality of our weight settings. The pixel-wise F1 score results are shown in Table 16. From the results, as the edge loss weight increases, the overall model performance improves. However, the overall fluctuation is slight, which demonstrates the training stability of our method.

Table 16: Ablation study on the weights of L_{seg} and L_{edg}.

$W_{\mathcal{L}_{seg}}$	$W_{\mathcal{L}_{edg}}$	CAS	COL	COV	NIST	Avg.F1
0.65	0.35	0.845	0.933	0.636	0.440	0.714
0.5	0.5	0.852	0.936	0.646	0.452	0.722
0.35	0.65	0.865	**0.944**	0.639	0.456	0.726
0.2 (Ours)	0.8 (Ours)	**0.873**	0.939	**0.659**	**0.480**	**0.738**

M Limitations

Although MAMQ-Net achieves strong detection performance, it still has several limitations. First, the full model is not the most lightweight, which may hinder deployment on resource-constrained devices. Second, the model relies on pixel-level annotations for training, which are costly to obtain at scale. Finally, through further analysis, we find that the localization accuracy of MAMQ-Net decreased in the following two scenarios, and we provide possible reasons: (1) heavy compression, resizing, or social-network re-encoding. Like most noise-based forensic methods, performance may degrade when images undergo strong post-processing which can attenuate the noise fingerprints exploited by the model. (2) Small-scale manipulations. The successive downsampling operations in the encoder dilute the weak noise residual signals from tiny tampered regions. Combined with extreme pixel-level class imbalance, the gradient contribution from small targets is negligible during backpropagation. These are also areas that we need to improve in the future.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims made in the abstract and introduction accurately reflect the paper's contributions and scope

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses the limitations of the work, such as the model requiring more computational resources.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The paper provides the full set of assumptions and a complete (and correct) proof.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have explicitly detailed the experimental setup in the paper and will open-source the code.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We provide detailed experimental settings in this paper and will open-source our code.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide detailed experimental settings in this paper and will open-source our code.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: All experiments were conducted only once, but given the sufficiently large scale, the results can still serve as a valuable reference.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide detailed experimental settings in this paper.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: All experiments in this paper are conducted using existing public datasets and comply with ethical standards.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the proposed paper's positive and negative societal impacts.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: This paper posed no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All resources used in this paper are open-source and publicly available.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We will provide the detailed document along with the open-source code.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development in this research does not involve LLMs as components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.