DG-Force: Disentangling and Gathering Forensic Cues is Needed for Image Manipulation Localization

Yong Yao[1,2*], Zhenyu Cui[3*], Lei Chen[3], Jiwen Lu[3], and Jiahuan Zhou[1†]

[1] Wangxuan Institute of Computer Technology, Peking University, Beijing, China
jiahuanzhou@pku.edu.cn
[2] Jilin University, Changchun, Jilin, China
yaoyong23@mails.jlu.edu.cn
[3] Department of Automation, Tsinghua University, Beijing, China
cuizhenyu@mail.tsinghua.edu.cn, {leichenthu,lujiwen}@tsinghua.edu.cn

Abstract. Image Manipulation Localization (IML) aims to achieve pixel-level localization of locally manipulated images. Its core challenge is how to distinguish the inconsistencies between authentic and tampered areas. To this end, existing IML methods typically exploit patch-level and edge-level forensic cues for accurate localization. However, these methods extract forged information at both levels with fixed parameters, leading to interference from real information that is detrimental to localization as more realistic manipulated textures are enhanced by the upgrading of tampering techniques. To tackle the above issue, this paper proposes a **D**isentangling and **G**athering **For**ensic **C**u**e**s method, called **DG-Force**, which explicitly disentangles and adaptively aggregates region and edge cues to preserve the most informative evidence. Specifically, we propose a Patch-based Forensic Disentangling (PFD) module and an Edge-based Forensic Disentangling (EFD) module to decompose forensic cues of different granularities to explicitly suppress the imbalance between forensic traces in both patch-level and edge-level areas. In addition, a Multi-scale Forensic Transfer (MFT) module is further designed to aggregate and balance multi-granularity information through the interaction across different scales for robust key inconsistencies in forensic traces discovery. Extensive experiments on multiple benchmarks demonstrate the superiority of our method in the image manipulation localization task.

Keywords: Image Manipulation Localization · Digital Image Forensics

1 Introduction

Image manipulation localization (IML) aims to achieve pixel-level localization of locally manipulated image by diverse tampering techniques [19, 22]. Its core

* Equal contribution.
† Corresponding author

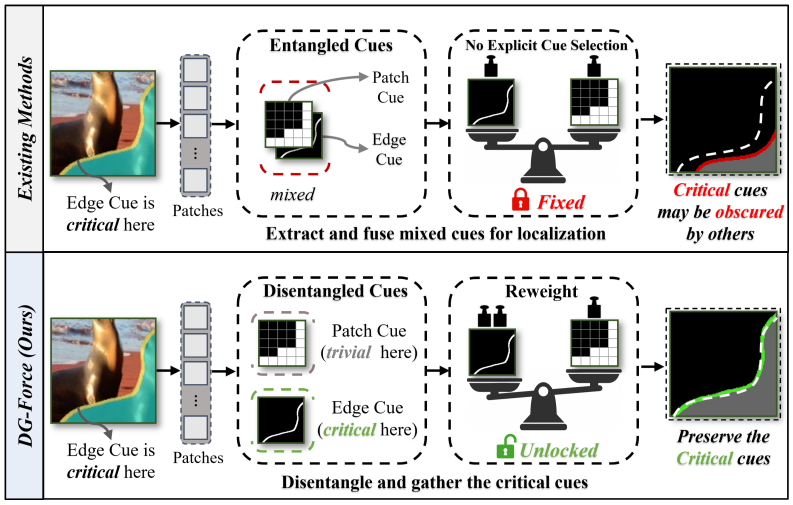


Fig. 1: Existing methods directly extract entangled forensic cues and treat them equally across different forgery scenarios. Our approach first disentangles these cues and then adaptively aggregates them according to their importance.

challenge lies in how to distinguish the low-level inconsistencies between authentic and tampering areas in images forged by different manipulation methods, *i.e.*, forensic traces. To this end, patch-level IML methods devote to locate manipulated regions through identifying inconsistencies in texture [16, 22, 28, 30], or noise levels [6, 26, 27], across distinct regions, while edge-level IML methods capture abrupt changes of the pixels distributed along the outline [8] or the anomalies derived by discontinuous boundaries [42], etc. When it comes to various forensic traces brought by different tampering techniques, some recent advancements have combined the aforementioned two aspects and have also made progress [40, 55].

To achieve simultaneous adaptation and detection to diverse types of image tampering techniques [5, 17, 19, 21, 34, 48, 54], existing methods [30, 55] typically employ IML models with constant parameters to extract forensic cues distributed across edges and patches. However, with the continuous upgrading of tampering techniques, increasingly realistic forged object textures and edges suppress the extraction of key forensic cues. Specifically, as shown in Fig. 1, when the quality of manipulated textures in image patches is enhanced by updated tampering techniques, employing models with constant parameters to merge both traces within patches and edges inevitably results in the merged traces containing more authentic texture information, suppressing the proportion of potentially but crucial forensic cues distributed at the edges. Furthermore, due to the agnostic nature of manipulation techniques in the inference, using patch or texture information at a single scale is typically insufficient to determine the amount of forensic traces. As a result, existing methods often exacerbate the imbalance

of forensic traces out of manipulated regions, compromising the integrity and robustness of image localization. As shown in Fig. 2, the vanilla IML model fails to extract key forensic cues when confronted with novel manipulations, which leads to imbalanced evidence and inaccurate localization.

To address the above issues, we propose a Disentangling and Gathering Forensic Cues (DG-Force) method for image manipulation localization. It starts by separating patch-level and edge-level forensic cues and then adaptively gathers them based on the amount of forensic traces in distinct regions. Specifically, we designed a Dual Forensic Disentangling and Gathering (DFDG) module, which contains a patch-based and an edge-based forensic disentangling to decompose forensic traces of different granularities. Furthermore, for each image token, we introduced a lightweight forgery amount evaluation network to assess the forensic traces of different manipulations, which is used to adaptively gather patch-level and edge-level forensic cues. Therefore, it can dynamically adjust the proportion of different forensic cues under different downstream tasks, thus overcoming the catastrophic imbalance problem of forensic traces. Finally, to further improve the capacity in identifying agnostic manipulation types, we also introduced a Multi-scale Forensic Transfer (MFT) module to aggregate forensic cues at different perspectives, allowing the collaborative discovery of key inconsistencies in forgery information at different granularities.

In summary, our main contributions are as follows:

- A Disentangling and Gathering Forensic Cues (DG-Force) method is proposed to tackle the image manipulation localization challenge, which aims to alleviate the catastrophic imbalance problem of forgery information distributed at different granularities.
- A Patch-based Forensic Disentangling (PFD) module and an Edge-based Forensic Disentangling (EFD) module to achieve decomposition of forensic cues of different granularities, which explicitly suppresses the imbalance between forensic traces in both patch-level and edge-level areas.
- A Multi-scale Forensic Transfer (MFT) module is introduced to aggregate and balance multi-granularity information through the interaction across different scales for robust key inconsistencies in forensic traces discovery.

2 Related Work

2.1 Image Forgery Techniques

Early image manipulations were typically created through manual editing, including copy-move, splicing, and inpainting, often followed by blending, smoothing, or compression to better match the surrounding context. More recently, AI-based image tampering has advanced rapidly, and generative approaches for localized edits have become increasingly common. These methods edit selected regions using GAN-based models [4, 20], diffusion-based models [3, 29, 32, 36], or instruction-driven multimodal editors [33, 37], enabling insertion, removal, or

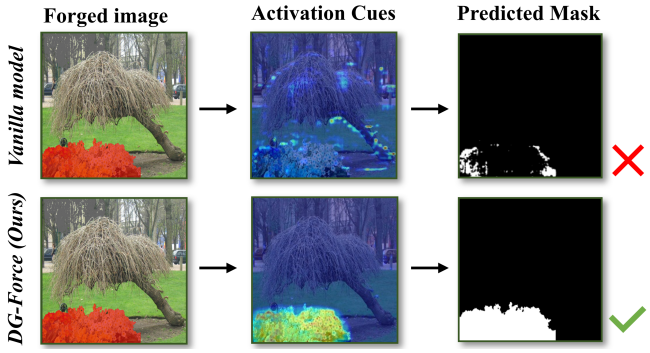


Fig. 2: The visualizations compare the activated cues: the red mask denotes the forged region, and the model-activated cues are visualized using Grad-CAM [39].

replacement while producing realistic textures and visually coherent transitions. These diverse manipulation techniques pose new challenges to the generalization ability of image manipulation localization.

2.2 Image Manipulation Localization Methods

Patch-level IML Methods

Patch-level IML methods capture texture or color anomalies between manipulated regions and authentic content. Since many artifacts appear as local texture inconsistencies, several approaches [1, 14, 38, 41, 49] examine patch-level mismatches to extract cross-region forensic cues, and some further consider finer artifacts such as incorrect color interpolation [22] to localize abnormal areas. Because a single spatial scale is difficult to capture multi-scale traces with varied shapes, several studies adopt multi-scale designs [26,28,50] that fuse texture cues across resolutions, improving sensitivity to manipulated textures. Despite favorable performance in many cases, patch-level methods still struggle to extract reliable forensic cues as the increasingly realistic textures.

Edge-level IML Methods

Motivated by the limitations of patch-level methods, edge-level IML methods emphasize object boundaries, since local manipulations often leave more visible traces along clear edges. Some approaches [30, 35, 51] predict an outline mask of the forged region and add an auxiliary boundary loss during training, encouraging the model to focus on boundary-related artifacts. Other methods [25, 46] do not directly input RGB images, but instead use signals processed by High-pass filters as input, which retain more edge-level cues while suppressing content-dependent textures and other irrelevant details.

Mixed IML Methods

Mixed IML methods combine complementary cues, such as patch-level and edge-level evidence, by leveraging both texture information and additional signals to capture manipulation traces more reliably. Some methods [6, 16, 47, 52]

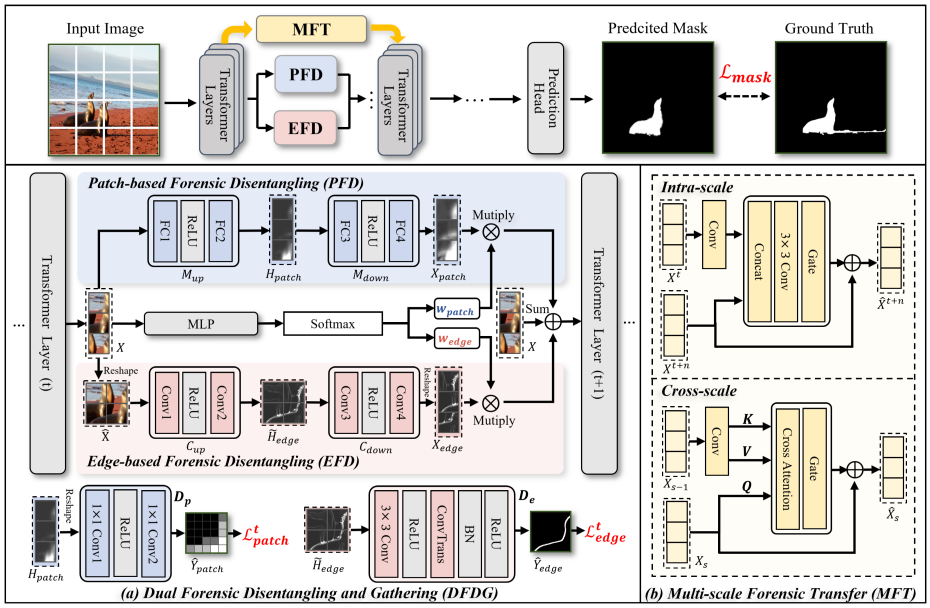


Fig. 3: Overall architecture of DG-Force. The top illustrates the main pipeline of our method, while the bottom shows the details of (a) the Dual Forensic Disentangling and Gathering (DFDG) module and (b) the Multi-scale Forensic Transfer (MFT) module.

apply SRM [9] or BayarConv [2] to extract residual responses, then fuse them with RGB features for localization. Other approaches [13,42,55] pair RGB inputs with frequency-domain transforms such as DCT [10], strengthening edge-level evidence and encouraging the model to account for both texture inconsistencies and boundary discontinuities. Additional methods [12, 40, 53] use dedicated feature extractors to obtain semantics-agnostic forensic cues from specialized low-level signals and combine them with texture information for localization.

Different from these methods, we explicitly disentangle patch-level and edge-level forensic cues through supervised patch and edge branches, and then adaptively aggregate the most informative evidence across spatial regions, thereby improving the robustness across diverse manipulation types.

3 Method

We propose DG-Force (Fig. 3) for image manipulation localization. DG-Force separates patch-based and edge-based forensic cues and adaptively aggregates them across scales. Sec. 3.1 describes the multi-scale Transformer backbone; Sec. 3.2 introduces Dual Forensic Disentangling and Gathering (DFDG); Sec. 3.3 presents Multi-scale Forensic Transfer (MFT); Sec. 3.4 details the overall loss.

3.1 Preliminaries: Multi-Scale Transformer

We use SegFormer [45] as the backbone. Its hierarchical multi-scale architecture preserves fine details while modeling long-range context, which suits pixel-level manipulation localization. SegFormer includes a four-stage encoder $\{\mathcal{E}_l\}_{l=1}^{4}$ and a lightweight decoder $\mathcal{D}$. Each encoder stage starts with an embedding layer and then applies a stack of Transformer blocks. The first stage takes the input image, and each subsequent stage processes the output from the previous stage. Formally, stage l produces tokens as

$$X_l = \mathcal{E}_l\big(X_{l-1}\big), \quad l = 1, 2, 3, 4, \tag{1}$$

where $X_l = \{x_i\}_{i=1}^{N_l}$, $x_i \in \mathbb{R}^{d_l}$, N_l is the number of tokens, and d_l is the channel dimension in stage l. In the end, we keep the outputs from all stages, i.e., $\{X_l\}_{l=1}^{4}$, and pass them to the decoder $\mathcal{D}$ for multi-scale fusion. The fused representation is then fed to a task-specific prediction head to produce the final output.

3.2 Dual Forensic Disentangling and Gathering

To address shifts in cue strength across manipulation scenarios and reduce the risk of discarding useful evidence, we introduce Dual Forensic Disentangling and Gathering (DFDG), illustrated in Fig. 3(a). DFDG includes two branches, Patch-based Forensic Disentangling (PFD) and Edge-based Forensic Disentangling (EFD), which decompose the uncoupled input features into patch-level and edge-level cues, respectively. A lightweight gating network then adaptively combines these cues based on their estimated importance. DFDG can be implemented between any Transformer layers at each scale. For clarity, we describe the module in a single-scale setting in this section.

Patch-based Forensic Disentangling

The Patch-based Forensic Disentangling (PFD) branch aims to capture texture inconsistencies at the patch level and determine whether a local image patch contains forged content, using coarse forensic cues derived from individual patches to support image manipulation localization.

Specifically, PFD includes a patch-level cue extractor $M_{\text{down}}(\cdot)$, a restoration layer $M_{\text{up}}(\cdot)$, and a patch decoder $D_p(\cdot)$. Given a token sequence from an arbitrary Transformer layer t, denoted as $X = \{x_i\}_{i=1}^{N}$ with $X \in \mathbb{R}^{N\times d}$, PFD processes these tokens independently. It first maps each token to a lower-dimensional representation and then projects it back to dimension d, producing disentangled patch-level tokens X_{patch}:

$$H_{\text{patch}} = M_{\text{down}}^{t}(X), \quad H_{\text{patch}} \in \mathbb{R}^{N\times(d/r_p)}, \tag{2}$$

$$X_{\text{patch}} = M_{\text{up}}^{t}(H_{\text{patch}}), \quad X_{\text{patch}} \in \mathbb{R}^{N\times d}, \tag{3}$$

where r_p is the reduction ratio. $M_{\text{down}}^{t}(\cdot)$ and $M_{\text{up}}^{t}(\cdot)$ are two-layer MLPs applied to all tokens in the t-th Transformer layer independently and in parallel.

To guide the PFD branch to extract accurate patch-level forensic cues, we use a soft patch label as supervision. For each token, its patch label is defined as the fraction of forged pixels within the corresponding image patch Ω_i with $|\Omega_i|$ pixels:

$$y_i^{\text{patch}} = \frac{1}{|\Omega_i|} \sum_{u \in \Omega_i} \text{mask}(u), \tag{4}$$

where $\text{mask}(u)$ is the binary pixel-level ground truth (1 for forged, 0 for authentic). The complete set of patch labels is $Y_{\text{patch}} = \{y_i^{\text{patch}}\}_{i=1}^{N}$, where each value lies in $[0, 1]$ and reflects the forgery proportion of the patch.

PFD then predicts patch labels from the intermediate features using a lightweight decoder for patch-level branch:

$$\hat{Y}_{\text{patch}} = D_{\text{p}}(H_{\text{patch}}), \tag{5}$$

and the loss for the PFD branch at Transformer layer t is

$$\mathcal{L}_{\text{patch}}^{t} = \text{BCE}\big(Y_{\text{patch}}, \hat{Y}_{\text{patch}}\big), \tag{6}$$

where $\text{BCE}(\cdot, \cdot)$ denotes the binary cross-entropy loss. All layers within the same scale share one patch decoder, while different scales use separate decoders.

Although PFD processes all tokens in a layer in one forward pass, each token is handled independently, with no interaction across tokens. This design keeps the disentangled patch-level cues aligned with their local image regions, without influence from neighboring tokens or global context.

Edge-based Forensic Disentangling

The Edge-based Forensic Disentangling (EFD) branch aims to capture discontinuous boundaries and abrupt pixel inconsistencies around object regions. In contrast to PFD, which processes each token independently, EFD applies convolution operations on the entire 2D feature map to model local neighborhood inconsistencies and produce fine-grained, boundary-sensitive forensic cues.

Given a token sequence $X \in \mathbb{R}^{N \times d}$, we reshape it into a 2D feature map $\tilde{X}$, where $N = h \times w$ is the spatial resolution at stage t. EFD includes an edge-cue extractor $C_{\text{down}}(\cdot)$, an edge-cue restorer $C_{\text{up}}(\cdot)$, and an edge decoder $D_e(\cdot)$. The reshaped feature map is processed as

$$\tilde{H}_{\text{edge}} = C_{\text{down}}^{t}(\tilde{X}), \qquad \tilde{H}_{\text{edge}} \in \mathbb{R}^{h \times w \times (d/r_e)}, \tag{7}$$

$$\tilde{X}_{\text{edge}} = C_{\text{up}}^{t}(\tilde{H}_{\text{edge}}), \qquad \tilde{X}_{\text{edge}} \in \mathbb{R}^{h \times w \times d}, \tag{8}$$

where r_e is the channel-reduction ratio. $C_{\text{down}}(\cdot)$ and $C_{\text{up}}(\cdot)$ are implemented with several 3×3 convolutional blocks. By using convolutional receptive fields, EFD aggregates information from local neighborhoods rather than only relying on a single token, which helps capture edge-level cues more completely. Finally, we reshape $\tilde{X}_{\text{edge}}$ back to $X_{\text{edge}} \in \mathbb{R}^{N \times d}$, which is spatially aligned with the patch-level tokens X_{patch}.

For supervision, we derive a binary boundary mask $Y_{\text{edge}} \in \{0,1\}^{H\times W}$ from the manipulation mask. The edge decoder $D_e(\cdot)$ first refines the compressed feature $\tilde{H}_{\text{edge}}$ with a 3×3 convolution, and then progressively upsamples it with several transposed-convolution blocks to the target resolution, producing the predicted edge mask:

$$\hat{Y}_{\text{edge}} = D_e(\tilde{H}_{\text{edge}}), \tag{9}$$

and the EFD loss at layer t is

$$\mathcal{L}^t_{\text{edge}} = \text{BCE}\big(Y_{\text{edge}}, \hat{Y}_{\text{edge}}\big). \tag{10}$$

This supervision encourages the EFD branch to focus on boundary-sensitive edge-level cues, which complement the patch-level forensic cues.

Adaptively Gathering Forensic Cues

Forensic traces may be obscured by increasingly realistic texture information, leading to imbalanced forensic cues across diverse manipulation scenarios. To alleviate this issue, we use a lightweight evaluator that estimates the amount of forgery information for each token to determine which cue type carries more critical evidence based on the disentangled representations. Formally, given tokens $X \in \mathbb{R}^{N\times d}$ together with disentangled patch tokens X_{patch} and edge tokens X_{edge}, we compute per-token scores $\text{Z} \in \mathbb{R}^{N\times 2}$ using an MLP applied to X. We normalize these scores with Softmax to obtain weights $\text{W} \in \mathbb{R}^{N\times 2}$, whose two entries for each token sum to 1. Let $\text{W} = [w_{\text{patch}}, w_{\text{edge}}]$ denote the two columns of weights, which denote the strength of patch-level and edge-level cues. Finally, the original image tokens, patch tokens, and edge tokens are aggregated as

$$\tilde{X} = X + w_{\text{patch}} \odot X_{\text{patch}} + w_{\text{edge}} \odot X_{\text{edge}}, \tag{11}$$

where $\tilde{X}$ denotes enriched tokens, and $\odot$ is element-wise multiplication.

Because $w_{\text{patch}} + w_{\text{edge}} = 1$ for each token, the fusion adapts between the two cue types. When $w_{\text{patch}} \to 1$ and $w_{\text{edge}} \to 0$, $\tilde{X}$ relies mainly on patch-level cues and favors region-wise texture inconsistencies. When $w_{\text{edge}} \to 1$ and $w_{\text{patch}} \to 0$, the fusion emphasizes boundary cues and reduces the influence of noisy patch responses. This token-wise gating strengthens informative evidence and downweights weaker signals under different manipulation types, improving both stability and accuracy for image manipulation localization.

After reweighting and fusion, the enriched tokens are passed to the next Transformer layer so that subsequent feature extraction operates on adaptively aggregated cues across regions and manipulation scenarios.

3.3 Multi-scale Forensic Transfer

Given manipulation traces can appear at different sizes, single-scale traces are typically insufficient to recover comprehensive forensic cues. To address this, we propose the Multi-scale Forensic Transfer (MFT), shown in Fig. 3(b), which aggregates cues from multiple perspectives and collaboratively discovers key inconsistencies across layers and scales. MFT operates on both patch and edge tokens through two components: intra-scale transfer and cross-scale transfer.

Intra-scale Transfer

Intra-scale transfer carries patch/edge forensic tokens from shallow Transformer layers to deeper ones within the same stage, helping preserve useful evidence as features become more abstract.

Concretely, we inject information from the first n layers into the last n layers. For patch/edge tokens X^t at a shallow layer t, MFT transfers them to the tokens X^{t+n} at a deeper layer via

$$\hat{X}^{t+n} = X^{t+n} + G^{t+n} \odot \Phi\big(\big[\,\phi(X^t),\, X^{t+n}\,\big]\big)\,, \tag{12}$$

where $[\cdot,\cdot]$ denotes concatenation, $\odot$ is element-wise multiplication, $\phi(\cdot)$ and $\Phi(\cdot)$ are convolutional blocks, and G^{t+n} is a learned residual gate that modulates the injected signal to avoid overly aggressive aggregation that might corrupt valid content in target tokens. In this way, the tokens passed to the downstream DFDG modules retain richer forensic evidence.

Cross-scale Transfer

Cross-scale transfer propagates cues along SegFormer's bottom-up hierarchy, passing evidence from early high-resolution stages to later coarser stages while preserving local details for localization.

Specifically, X_{s-1} is taken from the last few layers of the preceding higher-resolution stage, and X_s comes from the early layers of the current stage. We first map X_{s-1} to the current resolution and channel dimension with a convolutional block $\psi(\cdot)$, and then fuse cross-scale tokens via a cross-attention block $\Psi(\cdot)$ that uses current-stage tokens as queries and the previous-stage tokens as keys and values. Finally, a gated residual is applied:

$$\hat{X}_s = X_s + G_s \odot \Psi(X_s, \psi(X_{s-1})). \tag{13}$$

After intra-scale transfer and cross-scale transfer, the updated patch and edge tokens are adaptively fused by DFDG and then passed to the next Transformer layer, which allows subsequent layers to aggregate cues from richer, multi-scale, manipulation-aware representations.

3.4 Loss Function

We supervise three objectives: (i) manipulation mask, (ii) patch-level cues, and (iii) edge-level cues. The manipulation mask objective is optimized using a standard binary cross-entropy loss:

$$\mathcal{L}_{\text{mask}} = \text{BCE}(\hat{M}, M), \tag{14}$$

where $\hat{M}$ and M denote the predicted and ground-truth masks. Auxiliary patch/edge losses are computed at each stage $s \in \{1, \ldots, S\}$ and supervised Transformer layer $t \in \{1, \ldots, T\}$ and averaged over all. The final objective is:

$$\mathcal{L} = \alpha_1\, \mathcal{L}_{\text{mask}} + \frac{1}{ST} \sum_{s=1}^{S} \sum_{t=1}^{T} \big(\alpha_2\, \mathcal{L}_{\text{patch}}^{s,t} + \alpha_3\, \mathcal{L}_{\text{edge}}^{s,t}\big), \tag{15}$$

where $\mathcal{L}_{\text{patch}}^{s,t}$ and $\mathcal{L}_{\text{edge}}^{s,t}$ are BCE losses at layer t of stage s, and $\alpha_1, \alpha_2, \alpha_3$ are hyperparameters to balance the three terms.

4 Experiment

4.1 Experimental Setup

Training Setting

We follow the CAT-Net protocol [24] and implement our method within IMDLBenCo [31]. All images are resized to 512×512. We train for 150 epochs on four NVIDIA H800 GPUs, using a batch size of 12 per GPU and gradient accumulation of 2. We use AdamW with weight decay 0.05. The learning rate follows a cosine decay schedule, starting from 1×10^{-4} and decaying to 5×10^{-7}, with a warm-up of 2 epochs.

For the architecture, we insert PFD and EFD into the early Transformer layers of the first three stages (i.e., $3/4/6$ layers for stages $1/2/3$, respectively). For MFT, we use $[1, 1, 2]$ intra-scale fusion layers for these scales and add two cross-scale fusion layers across them. We set the loss weights for mask localization, patch supervision, and edge supervision to 2.0, 1.0, and 1.0, respectively.

Testing Setting

We primarily evaluate localization performance and robustness on four test datasets: CASIA v1 [7], COVERAGE [44], NIST16 [11], and Columbia [15], which is consistent with common practice [41, 55]. We additionally evaluate generalization to AI-generated tampering on CoCoGlide [12] and AutoSplice [18].

Evaluation Metrics

We comprehensively report pixel-level F1, pixel-level IoU, and pixel-level AUC to evaluate localization performance. Unless otherwise specified, all metrics are computed using a threshold of 0.5.

4.2 State-of-the-Art Comparison

We compare against PSCC-Net [26], CAT-Net [24], MVSS-Net [6], TruFor [12], IML-ViT [30], SparseViT [41], Mesorch [55], and OpenSDI [43]. All methods are trained and evaluated under the CAT-Net protocol using official implementations and recommended settings. For each method, we use the image resolution in its paper and apply the same data augmentations across all methods.

Localization Results

As shown in Table 1, DG-Force achieves the best average performance across the three metrics on all four datasets, outperforming existing methods by a clear margin. Relative to the strongest baseline, DG-Force improves the average F1, IoU, and AUC by 2.02%, 1.03%, and 1.8%, respectively. On individual datasets, DG-Force achieves the best performance on COVERAGE, Columbia, and NIST16, while remaining competitive on CASIAv1.

These performance gains can be attributed to the flexible cue disentangling-and-gathering mechanism, which allows the model to dynamically select and aggregate the most informative forensic cues within local regions of a forged image according to the manipulation characteristics of each sample, thereby maintaining strong adaptability across diverse manipulation scenarios. As texture content becomes increasingly realistic, forgery artifacts can be concealed, which

Table 1: Main results of image manipulation localization. Best average result is highlighted in **red**, and second-best in **blue**. Single-dataset best results are **bold**, and second-best underlined. Methods marked with † indicate results from Mesorch [55].

Method	Venue	COVERAGE			CASIAv1			Columbia			NIST16			AVG		
		F1	IoU	AUC	F1	IoU	AUC	F1	IoU	AUC	F1	IoU	AUC	F1	IoU	AUC
MVSS-Net† [6]	TPAMI'22	0.486	0.3886	0.8705	0.5832	0.4907	0.9115	0.7399	0.6721	0.9332	0.3363	0.2593	0.7900	0.5364	0.4527	0.8763
PSCC-Net† [26]	TCSVT'22	0.4475	0.3009	0.8838	0.6304	0.4999	0.9188	0.8841	0.8138	0.9457	0.3457	0.2971	0.8279	0.5769	0.4779	0.8941
CAT-Net† [23]	IJCV'22	0.4273	0.3875	0.9168	0.8081	0.7481	0.9804	0.915	0.8953	0.9457	0.2521	0.2125	0.8216	0.6006	0.5609	0.9161
TruFor† [12]	CVPR'23	0.4573	0.4149	0.9423	0.8176	0.7746	0.9742	0.8845	0.8593	0.8995	0.348	0.2964	0.8781	0.6269	0.5863	0.9235
IML-ViT [30]	arXiv'23	0.5619	0.4994	0.9457	0.7415	0.6736	0.9605	0.7075	0.6868	0.7203	0.2942	0.2587	0.8755	0.5763	0.5296	0.8755
SparseViT [41]	AAAI'25	0.5240	0.4721	0.9387	0.7143	0.6704	0.9652	0.9165	0.8983	0.9482	0.1511	0.1349	0.8452	0.5764	0.5439	0.9243
Mesorch† [55]	AAAI'25	0.5862	0.5360	0.9327	**0.8398**	**0.7877**	**0.9852**	0.8903	0.8766	0.9094	0.3921	0.3357	0.8883	**0.6771**	**0.6356**	**0.9289**
OpenSDI [43]	CVPR'25	0.4971	0.4357	0.9023	0.7765	0.7246	0.9723	0.8425	0.8196	0.8841	0.3612	0.3154	0.8782	0.6193	0.5738	0.9092
DG-Force(Ours)	–	**0.6163**	**0.5391**	**0.9553**	0.8063	0.7615	0.9796	**0.9383**	**0.9224**	**0.9632**	**0.4285**	**0.3611**	**0.8896**	**0.6973**	**0.6459**	**0.9469**

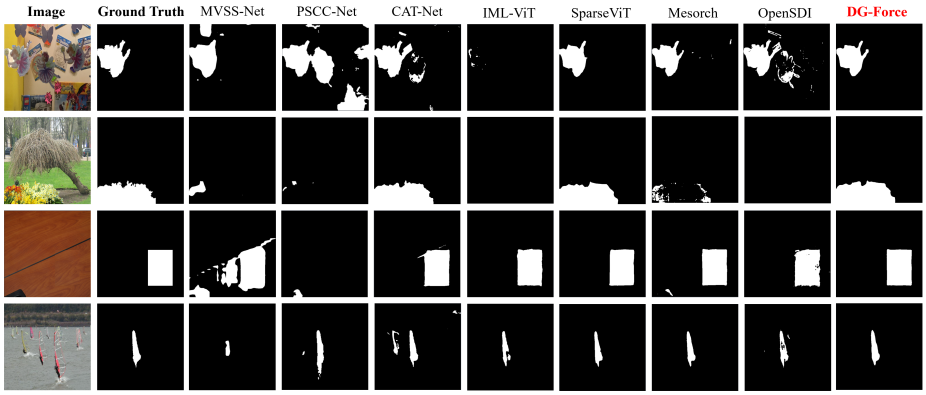


Fig. 4: Comparison of visualized prediction masks. The first two columns are the forged images and their corresponding tampered masks, the last column is our method.

suppresses subtle but critical edge cues and further exacerbates cue imbalance between authentic and manipulated regions. DG-Force mitigates this imbalance through an explicit cue disentangling design and a flexible weighting scheme that aggregates the most informative local cues under different manipulation scenarios, thereby improving overall localization performance.

Robust Performance

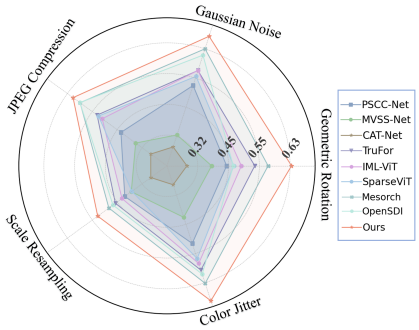
For robustness evaluation, we consider five common degradations in practical image processing: geometric rotation (Rot.), gaussian noise (Noise), JPEG compression (JPEG), scale resampling (Scale), and color variation (Color). For each degradation, we test six severity levels and report the average pixel-F1 over the four test datasets (CASIAv1, COVERAGE, NIST16, and Columbia). Details of each degradation setting are provided in the Appendix.

As shown in Tab. 2, DG-Force outperforms all competing methods by a clear margin. In particular, compared with the previous SOTA, it improves Geometric Rotation by 6.29%, Gaussian Noise by 3.3%, JPEG Compression by 2.27%,

Table 2: For the robustness comparison, we report the average pixel-F1 score across four test sets (CASIAv1, COVERAGE, NIST16, and Columbia). Detailed quantitative results are shown in (a), and the corresponding radar chart is shown in (b).

Method	Rot.	Noise	JPEG	Scale	Color
MVSS-Net	0.3809	0.3250	0.3548	0.3786	0.4198
PSCC-Net	0.4404	0.5241	0.4300	0.4117	0.5153
CAT-Net	0.2548	0.2540	0.2552	0.2550	0.2546
TruFor	0.5342	0.5708	0.5298	0.4552	0.5973
IML-ViT	0.4916	0.5710	0.5102	0.4252	0.5778
SparseViT	0.4662	0.5533	0.5246	0.3756	0.5633
Mesorch	0.5733	0.6326	0.5909	0.4832	0.6341
OpenSDI	0.4643	0.6152	0.5915	0.4659	0.6086
DG-Force	**0.6362**	**0.6656**	**0.6142**	**0.5276**	**0.6794**

(a) Average pixel-F1 robustness results.



(b) Radar chart of robustness results.

Table 3: Localization results on AI-based tampering image test sets: F1 and IoU using optimal thresholds. Best average result is highlighted in **red**, and second-best in **blue**.

Method	CoCoGlide			AutoSplice			AVG		
	F1*	IoU*	AUC	F1*	IoU*	AUC	**F1***	**IoU***	**AUC**
MVSS-Net	0.6378	0.5186	0.7898	0.6461	0.5109	0.6752	0.6419	0.5147	0.7325
PSCC-Net	0.6992	0.5868	0.8685	0.7882	0.6746	0.8799	0.7437	0.6307	0.8742
CAT-Net	0.4505	0.3299	0.6677	0.6524	0.5200	0.7556	0.5514	0.4249	0.7116
TruFor	0.7296	0.6287	0.8868	0.8120	0.7109	0.8979	**0.7708**	**0.6698**	**0.8923**
IML-ViT	0.6253	0.5069	0.8118	0.7242	0.5964	0.7897	0.6747	0.5516	0.8007
SparseViT	0.7198	0.6182	0.8778	0.8041	0.6985	0.8932	0.7619	0.6583	0.8855
Mesorch	0.7363	0.6380	0.8877	0.7975	0.6847	0.8936	0.7669	0.6613	0.8906
OpenSDI	0.7175	0.6120	0.8875	0.7981	0.6885	0.8714	0.7578	0.6502	0.8794
DG-Force(Ours)	**0.7762**	**0.6851**	**0.9091**	**0.8336**	**0.7335**	**0.9307**	**0.8049**	**0.7093**	**0.9199**

Scale Resampling by 4.44%, and Color Variation by 4.53%, indicating consistent robustness gains across perturbation types.

The superior robustness of DG-Force can be attributed to the mechanism of cue disentanglement and adaptive aggregation, which adapt well to diverse image degradations. Degradations can distort or obscure manipulation traces, which shifts the relative reliability of different cues and makes fixed extraction strategies less reliable. In contrast, by separating patch-level and edge-level cues, extracting them at multiple scales, and reweighting them based on their strength of forensic information, DG-Force preserves the most useful evidence under each degradation and improves robustness, which mitigates the imbalance among different cue types and ultimately improves robustness.

Generalization on AI-based Tampering

To assess generalization to unseen manipulation types, we evaluate DG-Force on two AI-generated datasets, CoCoGlide and AutoSplice. Since no AI-generated manipulations are included in the training data, IML methods often exhibit

substantial score calibration shifts under cross-domain evaluation, which causes fixed-threshold F1/IoU to underestimate the localization performance. We therefore report the best-threshold F1/IoU as an upper bound under ideal calibration, and additionally provide threshold-free AUC to measure pixel-level separability between manipulated and authentic regions.

As shown in Tab 3, DG-Force achieves the best average performance, improving F1, IoU, and AUC by 3.41%, 3.95%, and 2.76%, respectively. This result suggests that the feature disentanglement and adaptive balancing strategy maintain stable generalization to realistic AI-generated textures.

Visualization and Qualitative Analysis

Figure 4 presents qualitative examples across multiple manipulation types, including copy-move, splicing, and inpainting. For clarity, we show only the top 8 methods. DG-Force better preserves global geometry of manipulated objects while also recovering fine-grained boundary details, resulting in cleaner and more accurate localization masks. In contrast, prior methods often produce blurred boundaries, miss anomalies inside the manipulated region, or expand masks beyond the true extent of the tampering. A plausible explanation is that these methods extract entangled forensic cues without explicit separation, such that realistic textures or weakly informative signals can obscure the evidence needed to interpret boundaries. By disentangling patch-level and edge-level cues and reweighting them across regions, DG-Force keeps both interior consistency and boundary fidelity. Overall, these examples indicate that DG-Force captures forgery traces more reliably across diverse tampering types and datasets.

Model Complexity Analysis

As shown in Tab. 4, DG-Force uses 89.96M parameters and requires 93.65G FLOPs, ranking second only to SparseViT [41] in computational cost while achieving strong localization accuracy across diverse manipulation cases. This favorable balance between accuracy and efficiency largely stems from the lightweight disentangling-and-gathering modules, which improve cue extraction and fusion with limited overhead. Compared with Transformer-based baselines such as IML-ViT [30] and Mesorch [55], DG-Force achieves a more favorable balance between localization accuracy and computational complexity, delivering higher localization accuracy at a lower computational cost. Overall, DG-Force maintains practical efficiency alongside strong localization performance, which supports deployment in real-world scenarios.

4.3 Ablation Study

Table 5 reports the ablation results for DG-Force. Adding PFD and EFD branches individually improves F1 by 1.53% and 1.40%, indicating that disentangling patch-level and edge-level cues provides useful signals for localization. Using both branches with fixed equal weights brings only a small gain (0.21%), whereas introducing adaptive reweighting yields a much larger improvement of 2.29%. This suggests that fixed fusion can still cover critical evidence in some manipulation cases, while dynamic weighting better balances patch-level and edge-level cues

Table 4: Comparison of parameters and computational efficiency.

Method	**Parameters** (M)	**FLOPs** (G)
MVSS-Net	150.53	171.01
PSCC-Net	3.67	376.83
CAT-Net	116.74	137.22
TruFor	68.7	236.54
IML-ViT	91.74	136.23
SparseViT	50.35	46.22
Mesorch	85.75	124.93
OpenSDI	546.83	273.02
DG-Force(Ours)	89.96	93.65

Table 5: Ablation results for individual components, where Reweight corresponds to the adaptive weight adjustment mechanism in the DFDG. Base denotes the SegFormer.

Base	**PFD**	**EFD**	**Reweight**	**MFT**	**AVG-F1**
✓					0.6479
✓	✓				0.6632
✓		✓			0.6619
✓	✓	✓			0.6653
✓	✓	✓	✓		0.6882
✓	✓	✓	✓	✓	**0.6973**

and preserves the most discriminative information. Enabling MFT further improves performance by 0.91%, showing that multi-scale transfer helps by carrying fine-grained shallow evidence into deeper layers for pixel-level localization.

Overall, the full model achieves a total gain of 4.94% over the baseline. These results validate the contribution of each DG-Force module and show that cue disentanglement and adaptive gathering mitigate the imbalance among different manipulation types of forensic cues.

5 Conclusion

In this paper, we propose DG-Force for image manipulation localization task. DG-Force disentangles patch-level and edge-level forensic cues and adaptively aggregates them across scales, which helps mitigate cue imbalance. Extensive experiments and ablations demonstrate strong performance and highlight the contribution of each component.

One limitation of DG-Force is that the added multi-scale and disentanglement/aggregation modules make the computational cost sensitive to the architectural configuration. In practice, deployment may require adjusting the number of active layers and fusion operations to balance accuracy and efficiency.

In future work, we will extend this dynamic cue fusing mechanism to large pre-trained models and explore alternative fusion strategies that better leverage their priors to improve cross-domain generalization.

Acknowledgements

This work was a research achievement of National Engineering Research Center of New Electronic Publishing Technologies, and was supported by the National Natural Science Foundation of China (62376011), and the National Key R&D Program of China (2024YFA1410000).

References

1. Bappy, J.H., Simons, C., Nataraj, L., Manjunath, B.S., Roy-Chowdhury, A.K.: Hybrid lstm and encoder–decoder architecture for detection of image forgeries. IEEE Transactions on Image Processing **28**(7), 3286–3300 (Jul 2019). https://doi.org/10.1109/tip.2019.2895466, http://dx.doi.org/10.1109/TIP.2019.2895466
2. Bayar, B., Stamm, M.C.: A deep learning approach to universal image manipulation detection using a new convolutional layer. In: Proceedings of the 4th ACM workshop on information hiding and multimedia security. pp. 5–10 (2016)
3. Brooks, T., Holynski, A., Efros, A.A.: Instructpix2pix: Learning to follow image editing instructions (2023), https://arxiv.org/abs/2211.09800
4. Choi, Y., Choi, M., Kim, M., Ha, J.W., Kim, S., Choo, J.: Stargan: Unified generative adversarial networks for multi-domain image-to-image translation (2018), https://arxiv.org/abs/1711.09020
5. Cozzolino, D., Poggi, G., Verdoliva, L.: Efficient dense-field copy–move forgery detection. IEEE Transactions on Information Forensics and Security **10**(11), 2284–2297 (2015)
6. Dong, C., Chen, X., Hu, R., Cao, J., Li, X.: Mvss-net: Multi-view multi-scale supervised networks for image manipulation detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(3), 3539–3553 (2022)
7. Dong, J., Wang, W., Tan, T.: Casia image tampering detection evaluation database. In: 2013 IEEE China summit and international conference on signal and information processing. pp. 422–426. IEEE (2013)
8. Dua, S., Singh, J., Parthasarathy, H.: Detection and localization of forgery using statistics of dct and fourier components. Signal processing: image communication **82**, 115778 (2020)
9. Fridrich, J., Kodovsky, J.: Rich models for steganalysis of digital images. IEEE Transactions on information Forensics and Security **7**(3), 868–882 (2012)
10. Gonzalez, R.C.: Digital image processing. Pearson education india (2009)
11. Guan, H., Kozak, M., Robertson, E., Lee, Y., Yates, A.N., Delgado, A., Zhou, D., Kheyrkhah, T., Smith, J., Fiscus, J.: Mfc datasets: Large-scale benchmark datasets for media forensic challenge evaluation. In: 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW). pp. 63–72. IEEE (2019)
12. Guillaro, F., Cozzolino, D., Sud, A., Dufour, N., Verdoliva, L.: 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. pp. 20606–20615 (2023)
13. Guo, X., Liu, X., Ren, Z., Grosz, S., Masi, I., Liu, X.: Hierarchical fine-grained image forgery detection and localization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3155–3165 (2023)

14. Hao, J., Zhang, Z., Yang, S., Xie, D., Pu, S.: Transforensics: image forgery localization with dense self-attention. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15055–15064 (2021)
15. Hsu, Y.F., Chang, S.F.: Detecting image splicing using geometry invariants and camera characteristics consistency. In: 2006 IEEE international conference on multimedia and expo. pp. 549–552. IEEE (2006)
16. Hu, X., Zhang, Z., Jiang, Z., Chaudhuri, S., Yang, Z., Nevatia, R.: Span: Spatial pyramid attention network for image manipulation localization. In: European conference on computer vision. pp. 312–328. Springer (2020)
17. Huh, M., Liu, A., Owens, A., Efros, A.A.: Fighting fake news: Image splice detection via learned self-consistency. In: Proceedings of the European conference on computer vision (ECCV). pp. 101–117 (2018)
18. Jia, S., Huang, M., Zhou, Z., Ju, Y., Cai, J., Lyu, S.: Autosplice: A text-prompt manipulated image dataset for media forensics. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 893–903 (2023)
19. Kadha, V., Bakshi, S., Das, S.K.: Unravelling digital forgeries: A systematic survey on image manipulation detection and localization. ACM Computing Surveys **57**(12), 1–36 (2025)
20. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and improving the image quality of stylegan (2020), `https://arxiv.org/abs/1912.04958`
21. Kniaz, V.V., Knyaz, V., Remondino, F.: The point where reality meets fantasy: Mixed adversarial generators for image splice detection. Advances in neural information processing systems **32** (2019)
22. Kong, C., Luo, A., Wang, S., Li, H., Rocha, A., Kot, A.C.: Pixel-inconsistency modeling for image manipulation localization. IEEE Transactions on Pattern Analysis and Machine Intelligence (2025)
23. Kwon, M.J., Nam, S.H., Yu, I.J., Lee, H.K., Kim, C.: Learning jpeg compression artifacts for image manipulation detection and localization. International Journal of Computer Vision **130**(8), 1875–1895 (2022)
24. Kwon, M.J., Yu, I.J., Nam, S.H., Lee, H.K.: 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. pp. 375–384 (2021)
25. Li, H., Huang, J.: Localization of deep inpainting using high-pass fully convolutional network. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 8300–8309. IEEE/CVF (2019)
26. Liu, X., Liu, Y., Chen, J., Liu, X.: Pscc-net: Progressive spatio-channel correlation network for image manipulation detection and localization. IEEE Transactions on Circuits and Systems for Video Technology **32**(11), 7505–7517 (2022)
27. Liu, Y., Chen, S., Shi, H., Zhang, X.Y., Xiao, S., Cai, Q.: Mun: Image forgery localization based on m^3 encoder and un decoder. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 5685–5693 (2025)
28. Lou, Z., Cao, G., Guo, K., Yu, L., Weng, S.: Exploring multi-view pixel contrast for general and robust image forgery localization. IEEE Transactions on Information Forensics and Security (2025)
29. Lugmayr, A., Danelljan, M., Romero, A., Yu, F., Timofte, R., Gool, L.V.: Repaint: Inpainting using denoising diffusion probabilistic models (2022), `https://arxiv.org/abs/2201.09865`

30. Ma, X., Du, B., Jiang, Z., Hammadi, A.Y.A., Zhou, J.: Iml-vit: Benchmarking image manipulation localization by vision transformer. arXiv preprint arXiv:2307.14863 (2023)
31. Ma, X., Zhu, X., Su, L., Du, B., Jiang, Z., Tong, B., Lei, Z., Yang, X., Pun, C.M., Lv, J., et al.: Imdl-benco: A comprehensive benchmark and codebase for image manipulation detection & localization. Advances in Neural Information Processing Systems **37**, 134591–134613 (2024)
32. Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J.Y., Ermon, S.: Sdedit: Guided image synthesis and editing with stochastic differential equations (2022), https://arxiv.org/abs/2108.01073
33. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents (2022), https://arxiv.org/abs/2204.06125
34. Rao, Y., Ni, J.: A deep learning approach to detection of splicing and copy-move forgeries in images. In: 2016 IEEE international workshop on information forensics and security (WIFS). pp. 1–6. IEEE (2016)
35. Rao, Y., Ni, J.: Self-supervised domain adaptation for forgery localization of jpeg compressed images. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 15034–15043 (2021)
36. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models (2022), https://arxiv.org/abs/2112.10752
37. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E., Ghasemipour, S.K.S., Ayan, B.K., Mahdavi, S.S., Lopes, R.G., Salimans, T., Ho, J., Fleet, D.J., Norouzi, M.: Photorealistic text-to-image diffusion models with deep language understanding (2022), https://arxiv.org/abs/2205.11487
38. Salloum, R., Ren, Y., Jay Kuo, C.C.: Image splicing localization using a multi-task fully convolutional network (mfcn). Journal of Visual Communication and Image Representation **51**, 201–209 (Feb 2018). https://doi.org/10.1016/j.jvcir.2018.01.010, http://dx.doi.org/10.1016/j.jvcir.2018.01.010
39. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision. pp. 618–626 (2017)
40. Sheng, Z., Lu, W., Luo, X., Zhou, J., Cao, X.: Sumi-ifl: An information-theoretic framework for image forgery localization with sufficiency and minimality constraints. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 720–728 (2025)
41. Su, L., Ma, X., Zhu, X., Niu, C., Lei, Z., Zhou, J.Z.: Can we get rid of handcrafted feature extractors? sparsevit: Nonsemantics-centered, parameter-efficient image manipulation localization through spare-coding transformer. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 7024–7032 (2025)
42. Wang, J., Wu, Z., Chen, J., Han, X., Shrivastava, A., Lim, S.N., Jiang, Y.G.: Objectformer for image manipulation detection and localization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2364–2373 (2022)
43. Wang, Y., Huang, Z., Hong, X.: Opensdi: Spotting diffusion-generated images in the open world. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 4291–4301 (2025)

44. Wen, B., Zhu, Y., Subramanian, R., Ng, T.T., Shen, X., Winkler, S.: Coverage—a novel database for copy-move forgery detection. In: 2016 IEEE international conference on image processing (ICIP). pp. 161–165. IEEE (2016)
45. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. Advances in neural information processing systems **34**, 12077–12090 (2021)
46. Yang, C., Li, H., Lin, F., Jiang, B., Zhao, H.: Constrained r-cnn: A general image manipulation detection model. arXiv preprint arXiv:1911.08217 (2019)
47. Yang, C., Wang, Z., Shen, H., Li, H., Jiang, B.: Multi-modality image manipulation detection. In: 2021 IEEE International conference on multimedia and expo (ICME). pp. 1–6. IEEE Computer Society (2021)
48. Zanardelli, M., Guerrini, F., Leonardi, R., Adami, N.: Image forgery detection: a survey of recent deep-learning approaches. Multimedia Tools and Applications **82**(12), 17521–17566 (2023)
49. Zeng, K., Cheng, R., Tan, W., Yan, B.: Mgqformer: Mask-guided query-based transformer for image manipulation localization. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 6944–6952 (2024)
50. Zhou, J., Ma, X., Du, X., Alhammadi, A.Y., Feng, W.: Pre-training-free image manipulation localization through non-mutually exclusive contrastive learning. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 22346–22356 (2023)
51. Zhou, P., Chen, B.C., Han, X., Najibi, M., Shrivastava, A., Lim, S.N., Davis, L.: Generate, segment, and refine: Towards generic manipulation segmentation. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 13058–13065 (2020)
52. Zhou, P., Han, X., Morariu, V.I., Davis, L.S.: Learning rich features for image manipulation detection (2018), `https://arxiv.org/abs/1805.04953`
53. Zhu, J., Li, D., Fu, X., Shi, G., Xiao, J., Liu, A., Zha, Z.J.: A lottery ticket hypothesis approach with sparse fine-tuning and mae for image forgery detection and localization. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 10968–10976 (2025)
54. Zhu, X., Qian, Y., Zhao, X., Sun, B., Sun, Y.: A deep learning approach to patch-based image inpainting forensics. Signal Processing: Image Communication **67**, 90–99 (2018)
55. Zhu, X., Ma, X., Su, L., Jiang, Z., Du, B., Wang, X., Lei, Z., Feng, W., Pun, C.M., Zhou, J.Z.: Mesoscopic insights: orchestrating multi-scale & hybrid architecture for image manipulation localization. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 11022–11030 (2025)