IMPOSTOR: AN AGENT-CURATED BENCHMARK FOR REALISTIC AIGC MANIPULATION LOCALIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent advances in generative image editing have improved the realism and controllability of localized manipulation, raising new challenges for image manipulation detection and localization (IMDL). However, existing IMDL benchmarks have limitations in visual realism, manipulation diversity, and generator coverage, making it difficult to reflect recent trends in image manipulation. To address these limitations, we introduce Impostor, a high-quality AI-edited image manipulation localization dataset containing 100K manipulated images. Impostor is constructed by Disguise-Agent (DisAgent), a closed-loop agent framework that integrates scene perception, editing planning, manipulation execution, quality validation, and iterative reflection to automatically generate diverse and visually realistic manipulated images together with pixel-level masks, without requiring manual annotation. Moreover, Impostor contains images generated by seven recent AIGC models across three manipulation types and includes multiple manipulated regions, providing a more comprehensive benchmark for AIGC-based IMDL. Furthermore, we propose PhaseAware-Net (PANet), a semantic-forensic framework that introduces local phase modeling and Forensic-Semantic Consistency (FSC) learning to better localize semantically plausible yet forensically disrupted manipulated regions. Extensive experiments show that Impostor poses significant challenges to existing large vision-language models (LVLMs) and specialized IMDL methods, while PANet achieves superior performance on Impostor and multiple public benchmarks.

1 INTRODUCTION

Recent advances in generative image editing models (Labs et al., 2025; Wu et al., 2025) have made localized manipulations increasingly controllable and semantically consistent with the surrounding context, posing new challenges for image manipulation detection and localization (IMDL). However, existing IMDL benchmarks (Cai et al., 2026; Huang et al., 2025) cannot fully reflect recent trends in image manipulation. Specifically, as shown in Figure 1, existing datasets face three key limitations: (i) **limited visual realism**, where samples are often generated with simple editing instructions and the manipulated region could easily be found; (ii) **limited manipulation diversity**, where datasets cover only a few manipulation types and mainly focus on single-region edits; and (iii) **limited generator coverage**, where datasets only include limited generators. Therefore, existing benchmarks cannot fully assess whether IMDL methods remain effective in modern AIGC-based manipulation scenarios.

Moreover, to achieve accurate IMDL, extracting high-quality forensic cues, such as residual noise and frequency statistics, plays an important role in prior works (Dong et al., 2022; Kwon et al., 2022). However, existing works still face two limitations in AIGC-based manipulation scenarios. First, as suggested in (Chai et al., 2020; Zhong et al., 2024), forensic cues in manipulated regions may be disrupted, leading to weaker consistency with semantic representations. In contrast, forensic cues extracted from authentic regions are generally more consistent with their semantic representations. However, existing IMDL methods often simply fuse semantic and forensic representations without explicitly supervising such consistency. Second, many IMDL methods rely on global frequency-domain analysis, which overlooks local phase structures disrupted by AIGC editing and is less sensitive to the disruption of local forensic cues. Overall, these limitations weaken the model's ability to distinguish subtle and visually plausible manipulated regions generated by advanced AIGC models.

To address these limitations, on the one hand, we introduce Impostor, a high-quality benchmark comprising 100K manipulated images generated by recent AIGC models. Impostor is designed to

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

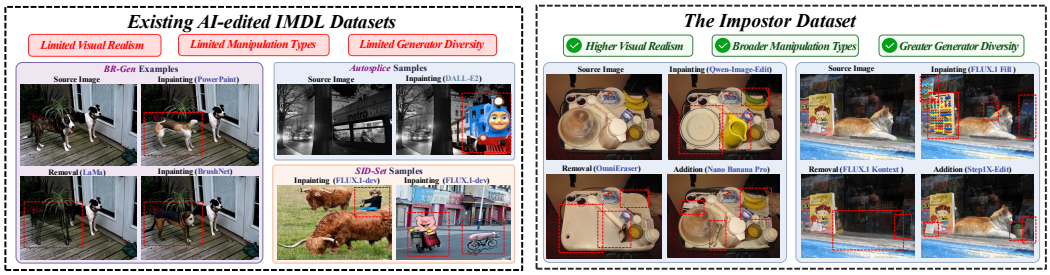

Figure 1: Comparison between Impostor and existing AIGC manipulation localization datasets. Impostor reflects better visual realism, manipulation diversity, and generator coverage, providing a more representative benchmark for modern AIGC-based image manipulations.

better approximate realistic localized editing scenarios from three aspects: **higher visual realism**, **greater manipulation diversity**, and **broader generator coverage**. For **visual realism**, Impostor is constructed by Disguise-Agent (DisAgent), a closed-loop agent framework that combines user-intent-aligned instruction generation with strict quality control. Specifically, the instruction generation stage provides high-quality editing prompts to guide AIGC models in performing localized edits. The quality control step further filters out samples with obvious visual artifacts or semantically implausible manipulations. In addition, all manipulated images undergo Turing-style human verification and are retained only if at least two annotators judge them as real. This step further improves the visual realism of the dataset from a human perception perspective. For **manipulation diversity**, as shown in Fig. 1, Impostor covers three representative manipulation types and multi-region manipulation scenarios. For **generator coverage**, Impostor integrates seven state-of-the-art image editing models, covering diverse visual characteristics and manipulation traces from recent AIGC editing models. In this way, Impostor provides a perceptually realistic and challenging benchmark for IMDL.

On the other hand, we propose PANet, a two-stream semantic-forensic framework for AIGC-based manipulation localization. Motivated by the above observation, PANet introduces a Forensic-Semantic Consistency (FSC) loss to explicitly supervise the consistency between semantic representations and forensic cues. FSC loss encourages authentic regions to produce stronger FSC responses, enabling subtle forensic cues to better guide the localization of visually plausible yet forensically disrupted manipulated regions. To support FSC learning with reliable forensic cues, we further introduce a Local Forensic Cue Extraction (LFCE) module. LFCE combines conventional residual cues with local phase modeling to capture fine-grained structural traces that global frequency analysis may miss. Finally, the semantic and forensic branches are fused through a Mutual Guided Cross Attention (MGCA) module, which enables multi-scale feature interaction for accurate manipulation localization.

In summary, our main contributions are as follows:

- We construct **Impostor**, a high-quality AIGC manipulation localization dataset curated by **DisAgent**, featuring higher visual realism, greater manipulation diversity, broader generator coverage, and automatically obtained pixel-level masks without manual annotation.
- We propose **PANet**, a two-stream framework that combines local phase modeling to capture subtle forensic traces with Forensic-Semantic Consistency (FSC) learning to explicitly model semantic-forensic correspondence, enabling accurate manipulation localization.
- Extensive experiments show that **Impostor** poses greater challenges to both LVLMs and specialized IMDL methods, while **PANet** achieves state-of-the-art performance on manipulation localization and exhibits remarkable generalization capacity.

2 Related Works

2.1 AI-generated Manipulation Localization Datasets

Detecting and localizing AI-generated image manipulations is essential for digital forensics and trustworthy visual communication. Traditional IMDL benchmarks (Dong et al., 2013; Ng et al., 2004; Guo et al., 2023) focus on conventional manipulations and are no longer representative of modern AI-driven image editing. To bridge this gap, CocoGlide (Guillaro et al., 2023) and AutoSplice (Jia et al., 2023) are among the first benchmarks to introduce diffusion-based image editing into the

IMDL setting. However, their scale and generator coverage remain limited. The advent of large foundation models enables automated and scalable benchmark construction. GIM (Chen et al., 2025) uses SAM (Ravi et al., 2025) and LLMs to construct a million-image benchmark, but its editing scenarios are restricted to salient objects from ImageNet (Deng et al., 2009). More recent datasets (Cai et al., 2026; Huang et al., 2025; Wang et al., 2025) incorporate advanced diffusion models and broaden manipulation settings, but primarily focus on single-region manipulations and still follow sequential construction pipelines. As shown in Figure 1, despite rapid progress, existing AIGC-oriented benchmarks still suffer from limited visual realism, manipulation diversity, and generator coverage, limiting their ability to fully assess IMDL methods under modern AIGC editing scenarios.

2.2 AI-generated Manipulation Localization Methods

To achieve accurate manipulation localization, many meticulously designed methods have been proposed. Early approaches (Dong et al., 2022; Liu et al., 2022b; Kwon et al., 2022) mainly rely on priors tailored to conventional manipulations, such as JPEG inconsistencies and boundary artifacts. However, high-quality AIGC edits often exhibit well-blended semantic boundaries, making these priors less effective. Consequently, recent research has shifted toward modeling subtler spectral and structural traces. Frequency-aware methods, such as F2D-Net (Li et al., 2025) and M2SFormer (Nam et al., 2025), enhance forensic representations through frequency-phase decomposition or DCT-based spectral modeling. More recent AIGC-oriented methods (Huang et al., 2026; Wang et al., 2026) further emphasize weak forensic trace enhancement and local-global contextual modeling, while RITA (Zhu et al., 2026) considers dependencies among composite manipulation operations. Despite this progress, existing methods mainly learn discriminative representations from frequency or forensic responses, rather than explicitly modeling spatially aligned local phase responses. Moreover, the forensic-semantic mismatch introduced by AIGC editing is rarely explicitly modeled or exploited for manipulation localization. These limitations motivate PANet, which combines local phase modeling with FSC learning for accurate AIGC manipulation localization.

3 Impostor Dataset

3.1 Source Dataset Selection

We select LVIS (Gupta et al., 2019) as the real image source, as its complex natural scenes and fine-grained mask annotations naturally support object-level manipulations across diverse semantic regions. This choice also mitigates shortcut bias in existing IMDL datasets (Chen et al., 2025; Cai et al., 2026) built from foreground-centric sources such as ImageNet, where models may localize salient objects rather than genuine manipulated regions.

3.2 DisAgent: A Closed-Loop Data Construction Framework

As shown in Figure 2, Impostor is constructed using DisAgent, a closed-loop agent framework designed to better simulate user editing behavior in practical scenarios. Inspired by the reasoning-and-acting paradigm of ReAct (Yao et al., 2022), DisAgent iteratively performs perception, planning, execution, validation, and reflection during dataset construction. Unlike prior agent-based construction pipelines such as EditStream (Wang et al., 2026), which do not explicitly propagate sample-level validation outcomes to subsequent planning, DisAgent closes this loop by reflecting on validation results, updating historical memory, and using this feedback to refine subsequent manipulation planning. Due to space limitations, detailed descriptions of each stage are provided in Appendix E.

User-intent-aligned instruction generation. Unlike existing AIGC manipulation benchmarks (Huang et al., 2025; Cai et al., 2026), in which manipulated images are often generated with simple instructions and limited scene understanding, realistic manipulation should better reflect real-world editing behavior. In practice, when using AIGC models for image editing, users typically first form an editing intent, identify a target region, and then formulate an instruction based on the local content and surrounding context. To better imitate this workflow, DisAgent first performs a **Perception** step to analyze the scene content and determine a suitable manipulation type, such as addition, removal, or inpainting. Then, conditioned on the selected manipulation type, DisAgent performs a **Planning** step to evaluate candidate segmentation masks provided by the source dataset based on their semantics, spatial layout, and surrounding context. It then selects suitable editing regions and generates the corresponding editing instructions. Since editing outcomes are highly sensitive to the

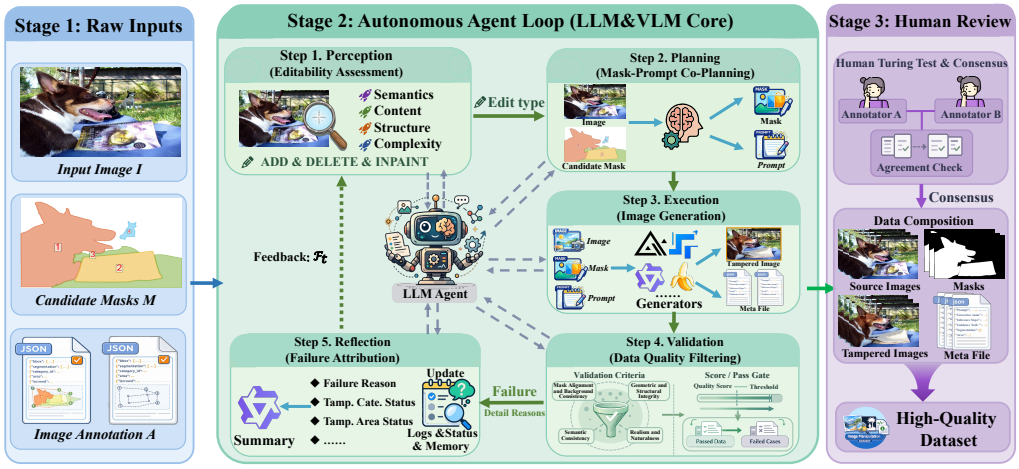

Figure 2: The generation pipeline of Impostor. An LVLM-based DisAgent iteratively perceives, plans, executes, validates, and reflects to generate diverse and high-quality manipulated images.

input instruction, effective prompts should be semantically plausible while remaining well aligned with the selected regions and surrounding scene context. To this end, we employ advanced LVLMs (OpenAI, 2025; Bai et al., 2025a) as the core decision-making module of DisAgent. Given the image and candidate masks, the LVLMs analyze fine-grained visual attributes of the target regions, such as shape, material, color, and texture, to support context-aware mask–instruction co-planning.

Multi-Generator Manipulation Construction. Existing AIGC manipulation benchmarks are often built with limited models, which reduces manipulation diversity and may encourage detectors to rely on generator-specific shortcuts. To address this issue, Impostor integrates seven state-of-the-art image editing models: six open-source models, including FLUX.1 Fill (Labs et al., 2025), FLUX.1 Kontext (Labs et al., 2025), PowerPaint (Zhuang et al., 2024), Step1X-Edit (Liu et al., 2025), Qwen-Image-Edit (Wu et al., 2025), and OmniEraser (Wei et al., 2025), and one closed-source model, Nano Banana Pro (Google DeepMind, 2025). Given the manipulation mask and editing instruction, DisAgent performs an **Execution** step to invoke suitable editing models for manipulated image generation. Importantly, DisAgent is agnostic to specific editing models, and new generators can be seamlessly integrated by extending the candidate model pool in the execution stage, enabling evolution with future AIGC editing models.

Quality Validation and Reflection. Ensuring both visual realism and spatial alignment is central to AIGC manipulation localization dataset construction. A valid sample should appear realistic while keeping the edited content strictly aligned with the annotated mask. However, generative editing models may fail to follow the instruction or modify regions outside the target mask. Therefore, after image generation, DisAgent performs a **Validation** step to assess each sample using four criteria: semantic consistency; mask alignment and background consistency; geometric and structural integrity; overall realism. Samples are then analyzed through a **Reflection** step, where the DisAgent summarizes failure attribution and other feedback signals to guide subsequent planning and generation. This closed-loop process reduces visually unrealistic samples and improves annotation reliability. Detailed validation criteria are provided in the Appendix E.5.

3.3 Human Verification and Dataset Finalization

To provide a high-quality and reliable dataset for the community, we further conduct the Turing-style human verification, where each manipulated image is independently reviewed by two annotators through a two-stage protocol. In the first stage, annotators are shown only the manipulated image and judge whether it appears to be a naturally captured photograph. In the second stage, images judged as natural are further checked for mask alignment with the manipulated region. A sample is retained only when both annotators approve its naturalness and mask alignment. Otherwise, this sample is discarded. Finally, human verification removes approximately 16% of the validated candidates, mainly due to visible artifacts or mask-region mismatches. It is important to emphasize that this human verification step serves only as an additional assurance step to ensure the high quality of Impostor, rather than a manual intervention in the data-generation pipeline.

Table 1: Summary of dataset attributes. For traditional datasets, splicing and copy-move are grouped as "Add". "Categories" denotes the number of basic semantic categories of manipulated objects, with details provided in Appendix C.3. Image-level detection and pixel-level localization results are evaluated on manipulated images only, with lower values indicating more challenging manipulations. * denotes the corresponding training source.

Type	DataSet	Dataset Scale					Manip. Type			Image-level Detection Rate				Pixel-level Localization IOU			
		Real	*Fake*	*Masks*	*Models*	*Categories*	*Inpaint*	*Add*	*Delete*	*Qwen3VL*	*FakeShield*	*Gemini3Pro*	*GPT-5.4*	*FakeShield*	*MVSS-Net*	*NFA-ViT*	*Sparse-ViT*
Manual	Columbia	183	180	1	-	30	✗	✓	✗	95.00	91.66	97.71	85.46	69.58	36.98	33.59	89.12
	CASIA V1	800	921	1	-	70	✗	✓	✗	44.07	92.01	64.24	47.58	56.74	42.67	21.51	66.81
	CASIA V2	7,491	5,123	1	-	145	✗	✓	✗	27.75	75.82	31.61	25.98	48.38	*66.72	16.55	*70.80
	PSCC-Net	81,910	194,829	1	-	80	✗	✓	✓	61.98	98.19	59.15	55.67	53.29	21.22	17.35	42.36
	SP COCO	-	200,000	1	-	80	✗	✓	✗	87.45	71.08	90.87	84.51	65.87	32.78	19.08	*83.51
	IMD2020	414	2,010	1-2	-	180	✗	✓	✗	79.17	93.42	83.75	71.40	63.87	22.59	19.04	*62.52
AIGC	CocoGlide	-	512	1-2	1	73	✓	✗	✗	67.18	100.00	51.95	54.70	42.96	22.56	64.21	27.19
	AutoSplice	2,273	3,621	1	1	134	✓	✗	✗	44.05	57.73	66.08	62.48	48.36	19.37	70.15	25.14
	SID-Set	100,000	100,000	1-2	1	300	✓	✗	✗	43.78	85.89	50.33	37.86	23.27	16.75	39.43	17.90
	BR-Gen	15,000	150,000	1	5	635	✓	✗	✓	50.29	30.51	65.49	46.20	25.35	9.85	*90.71	11.13
	Impostor	100,000	100,000	**1-9**	**7**	**2800**	✓	✓	✓	13.47	10.23	19.85	18.75	14.23	7.10	18.11	6.18

3.4 Benchmark Properties and Data Split

Impostor is a large-scale benchmark for AIGC manipulation localization comprising 100K source–manipulated image pairs, along with corresponding manipulation masks and generation metadata. Compared with prior datasets, Impostor is characterized by three main properties. First, Impostor exhibits higher visual realism. As shown in Table 1, existing LVLMs and forensic foundation models show substantially lower performance on Impostor, indicating that its manipulated images are more difficult to detect and providing additional evidence of higher visual realism. Second, Impostor provides greater manipulation diversity. Impostor covers three different manipulation types, with up to nine manipulated regions per image. Meanwhile, the manipulated objects span 2,800 categories, providing rich semantic diversity. Third, Impostor offers broader generator coverage. Impostor is constructed using seven advanced AIGC editing models, introducing diverse editing traces and visual distributions. These properties make Impostor a more comprehensive and challenging benchmark for modern AIGC manipulation localization.

For dataset partitioning, samples from each generator are divided into training and test subsets with a 7:1 split, and each source image along with all associated manipulated images is assigned to the same subset to prevent data leakage.

4 Methods

As shown in Figure 3, PANet extracts multi-scale semantic features and forensic features through semantic and LFCE-enhanced forensic branches, respectively. The MGCA then fuses semantic and forensic features across scales. Afterwards, an FPN decoder predicts a pixel-level mask together with an image-level score. During training, the FSC loss supervises forensic-semantic consistency responses at two deeper scales, turning the relative consistency gap between authentic and manipulated regions into discriminative localization guidance. Further empirical analyses in Appendix A provide evidence for the underlying mechanisms of FSC learning and LPCE-based local phase modeling.

4.1 Local Forensic Cue Extraction Module

To capture manipulation-sensitive local forensic cues, we introduce a Local Forensic Cue Extraction (LFCE) module. The LFCE module consists of two parallel blocks: a Residual Cue Extraction (RCE) block that extracts residual-domain traces, and a Local Phase Cue Extraction (LPCE) block that captures spatially aligned local phase traces. Their outputs are then fed into the forensic branch.

4.1.1 Residual Cue Extraction Block

The RCE block consists of three components: fixed SRM filters (Fridrich & Kodovsky, 2012) and two learnable constrained convolutions. Given an input image $I \in \mathbb{R}^{3\times H\times W}$, frozen SRM filters provide fixed high-pass residual priors, while constrained convolutions learn task-adaptive residual patterns. For each constrained convolution, its kernel $W \in \mathbb{R}^{K\times K}$ is constrained by setting the

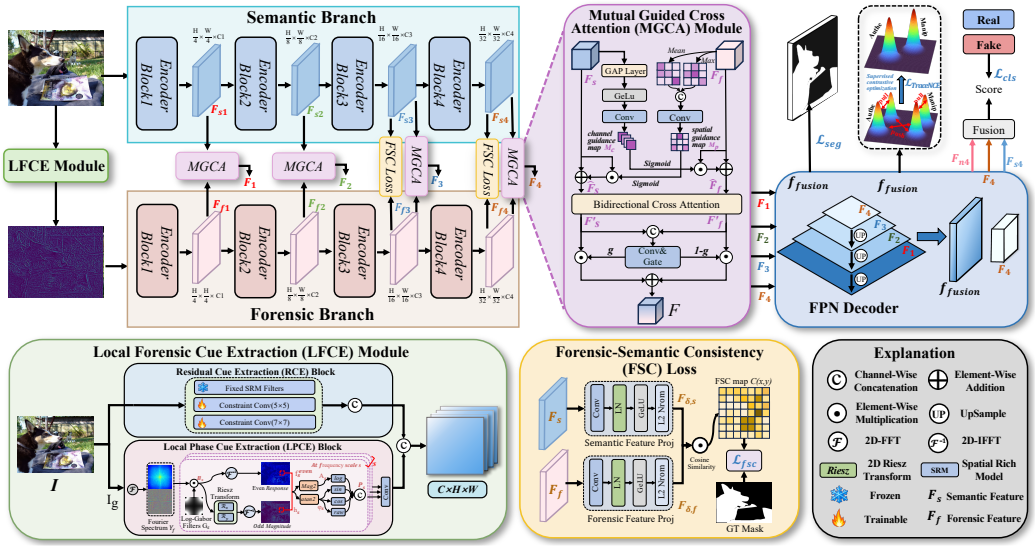

Figure 3: Overview of PANet. A semantic branch extracts multi-scale semantic features, while an LFCE-enhanced forensic branch extracts residual and spatially aligned local phase cues. These features are fused via MGCA and decoded by an FPN decoder. The FSC loss explicitly supervises forensic-semantic consistency, enabling subtle forensic cues to better guide manipulation localization.

center coefficient as $W_{c,c} = -\sum_{(u,v)\neq(c,c)} W_{u,v}$. This makes the kernel zero-sum, suppressing low-frequency content and emphasizing residual forensic cues.

4.1.2 Local Phase Cue Extraction Block

RCE mainly captures high-frequency residual statistics, which reflect the residual strength in local regions but provide limited information about spatial structure. Moreover, in realistic AIGC manipulations, the residual differences between manipulated and authentic regions can be subtle, further limiting the effectiveness of residual-domain forensic cues. To address this limitation, we introduce spatially aligned local phase representations. Local phase encodes structural coherence in edges, contours, and textures, making it more sensitive to disruptions introduced by localized manipulation.

We first convert the RGB image I to a grayscale image I_g and compute its Fourier transform as $Y_f = \mathcal{F}(I_g)$. At each frequency scale s, an isotropic Log-Gabor filter G_s extracts a frequency band, yielding the band-limited spectrum $B_s = Y_f \odot G_s$. Afterwards, we apply the inverse Fourier transform $\mathcal{F}^{-1}$ to B_s to obtain the even response f_s^{even} at scale s. In parallel, we apply the two-dimensional Riesz transform (Felsberg & Sommer, 2001) to B_s using the directional multipliers R_x and R_y, and then use $\mathcal{F}^{-1}$ to obtain two odd responses along the horizontal and vertical directions:

$$f_s^{even} = \mathrm{Re}\left(\mathcal{F}^{-1}(B_s)\right), \quad h_{x,s}^{odd} = \mathrm{Re}\left(\mathcal{F}^{-1}(B_s \odot \mathcal{R}_x)\right), \quad h_{y,s}^{odd} = \mathrm{Re}\left(\mathcal{F}^{-1}(B_s \odot \mathcal{R}_y)\right), \quad (1)$$

where $\mathcal{R}_x$ and $\mathcal{R}_y$ are defined as $\mathcal{R}_{x/y}(u,v) = -i\,\{u/v\}/\sqrt{u^2+v^2+\epsilon}$. We retain the real parts of the inverse Fourier outputs to remove negligible numerical imaginary components, and then aggregate the two directional odd responses into the odd magnitude $h_s = \sqrt{(h_{x,s}^{\text{odd}})^2 + (h_{y,s}^{\text{odd}})^2}$. Based on f_s^{even} and h_s, we compute the local amplitude A_s and the local phase ϕ_s as follows:

$$A_s = Mag2(f_s^{even}, h_s) \triangleq \sqrt{(f_s^{even})^2 + (h_s)^2}, \ \phi_s = \mathrm{atan2}(h_s, f_s^{even}), \quad (2)$$

where A_s and ϕ_s represent the local response strength and local phase structure at scale s, respectively. To jointly encode phase structure and response strength, we define a four-channel phase descriptor:

$$P_s = \mathrm{Concat}(\cos\phi_s,\ \sin\phi_s,\ \log(1+A_s),\ \phi_s)\,, \quad (3)$$

where $\cos\phi_s$ and $\sin\phi_s$ provide a bounded nonlinear encoding of the phase response, $\log(1+A_s)$ compresses the dynamic range of the response strength, and ϕ_s retains the raw phase value. Finally, we concatenate the phase descriptors from all frequency scales and process them with a lightweight convolutional module:

$$P = \mathrm{Concat}(P_1, P_2, \ldots, P_S), \qquad F^{\mathrm{LPCE}} = \mathrm{Conv}(P). \quad (4)$$

The resulting F^{LPCE} is fused with the RCE features and used as the input to the forensic branch.

4.2 Mutual Guided Cross Attention Module

To enhance the interaction between semantic and forensic features, we introduce a Mutual Guided Cross Attention (MGCA) module. Given the semantic feature $F_s \in \mathbb{R}^{B\times C\times H\times W}$ and forensic feature $F_f \in \mathbb{R}^{B\times C\times H\times W}$, the MGCA module generates a channel guidance map $M_c \in \mathbb{R}^{B\times C\times 1\times 1}$ from F_s and a spatial guidance map $M_p \in \mathbb{R}^{B\times 1\times H\times W}$ from F_f:

$$M_c = \sigma(\mathrm{Conv}(\mathrm{GeLU}(\mathrm{GAP}(F_s)))), \quad M_p = \sigma(\mathrm{Conv}(\mathrm{Concat}[\mathrm{Mean}(F_f), \mathrm{Max}(F_f)])), \tag{5}$$

where $\sigma(\cdot)$ denotes the sigmoid function. Consequently, M_p and M_c refine the semantic feature F_s and the forensic feature F_f, respectively, through residual modulation:

$$\hat{F}_s = F_s + F_s \odot M_p, \qquad \hat{F}_f = F_f + F_f \odot M_c. \tag{6}$$

After mutual guidance, we then apply standard bidirectional cross-attention to the guided features, producing F'_s and F'_f. Finally, a spatially adaptive gate $g \in \mathbb{R}^{B\times 1\times H\times W}$ is predicted from the concatenated attended features to perform location-wise fusion:

$$g = \sigma\big(\mathrm{Conv}_{1\times 1}\big(\mathrm{Concat}[F'_s, F'_f]\big)\big), \qquad F = g \odot F'_s + (1-g) \odot F'_f. \tag{7}$$

F is the final output of MGCA module. This location-wise fusion allows the model to emphasize the more informative branch at each spatial position, thereby improving manipulation localization.

4.3 Forensic-Semantic Consistency Loss

To explicitly model the consistency between semantic and forensic representations, we project semantic feature F_s and forensic feature F_f into a shared D-dimensional space and applies ℓ_2 normalization, producing $F_{\delta,s}$ and $F_{\delta,f}$. Their cosine similarity measures the local FSC score:

$$C(x,y) = \mathrm{CosSim}\left(F_{\delta,s}(:,x,y), F_{\delta,f}(:,x,y)\right), \qquad C \in \mathbb{R}^{B\times 1\times H\times W}, \tag{8}$$

where a larger $C(x,y)$ indicates stronger local FSC. Using the manipulation mask, we compute the average scores $\bar{C}_{real}$ and $\bar{C}_{fake}$ over authentic and manipulated regions, respectively. To encourage high FSC scores in authentic regions, we impose positive consistency supervision on authentic regions through $\mathcal{L}_{\mathrm{real}} = \mathbb{E}_{i\in\Omega_{\mathrm{real}}}\left[\mathrm{BCEWithLogits}(\tau C_i, 1)\right]$, where τ is a learnable scaling factor. Meanwhile, to enlarge the consistency gap between authentic and manipulated regions, we introduce a consistency separation loss $\mathcal{L}_{\mathrm{sep}} = \max\left(0, \delta - \left(\bar{C}_{\mathrm{real}} - \bar{C}_{\mathrm{fake}}\right)\right)$, where δ denotes the margin. The final FSC loss is $\mathcal{L}_{\mathrm{fsc}} = \mathcal{L}_{\mathrm{real}} + \mathcal{L}_{\mathrm{sep}}$. More details about FSC loss are provided in Appendix D.1.

4.4 Loss Function

The overall training objective consists of two primary losses, $\mathcal{L}_{\mathrm{seg}}$ and $\mathcal{L}_{\mathrm{cls}}$, and two auxiliary losses, $\mathcal{L}_{\mathrm{fsc}}$ and $\mathcal{L}_{\mathrm{TraceNCE}}$. Specifically, $\mathcal{L}_{\mathrm{seg}}$ supervises pixel-level manipulation localization using a hybrid Dice-BCE loss, while $\mathcal{L}_{\mathrm{cls}}$ supervises image-level real/fake classification with a BCE loss. In addition, $\mathcal{L}_{\mathrm{fsc}}$ enforces forensic-semantic consistency, and $\mathcal{L}_{\mathrm{TraceNCE}}$ improves region-level discrimination between authentic and manipulated features in the decoder feature space.

5 Experiments

5.1 Experimental Setup

Datasets and Evaluation Metrics. To evaluate the transferability of Impostor as a training source and verify the generalization capacity of PANet, we conduct both pixel-level localization and image-level detection experiments. For pixel-level localization experiments, beyond our Impostor dataset, we also conduct cross-domain experiments on CocoGlide (Guillaro et al., 2023), AutoSplice (Jia et al., 2023), SID-Set (Huang et al., 2025), and BR-Gen (Cai et al., 2026). We report F1-score (F1) and Intersection over Union (IoU) on manipulated images only. For image-level detection experiments, we follow the standard AIGCDetectBenchmark setting (Zhong et al., 2024), where models are trained on ProGAN and tested on multiple unseen synthetic image datasets. We report accuracy (Acc) on balanced sets containing equal numbers of real and manipulated images. For fair comparison, all methods are trained and evaluated under the same protocol.

Table 2: Pixel-level localization performance under in-domain and cross-domain evaluation. **Bold** and underline denote the best and second-best results, respectively.

Method	Year	Params	In-domain		Cross-domain								Average	
			Impostor		SID-Set		BR-Gen		CocoGlide		AutoSplice			
			IoU	F1	IoU	F1	IoU	F1	IoU	F1	IoU	F1	IoU	F1
MVSS-Net	2021 ICCV	146.9M	50.00	59.31	27.89	34.58	12.97	16.89	42.40	51.87	29.38	39.15	32.53	40.36
CAT-NET	2022 IJCV	114.3M	63.42	72.77	41.81	49.74	18.93	23.75	47.22	55.98	28.61	37.21	40.00	47.89
ObjectFormer	2022 CVPR	130.6M	54.19	65.53	24.82	31.62	30.30	35.75	52.35	62.11	68.14	77.91	45.96	54.58
TruFor	2023 CVPR	68.7M	51.17	59.89	22.31	27.40	16.85	23.13	36.21	50.66	16.69	22.62	28.65	36.74
Mesorch	2025 AAAI	85.8M	70.79	78.76	39.08	44.46	24.57	28.06	64.23	71.61	70.30	77.58	53.79	60.09
M2SFormer	2025 ICCV	28.1M	53.50	64.24	33.97	40.64	24.38	30.76	53.23	61.82	54.41	65.27	43.90	52.55
NFA-ViT	2026 AAAI	28.2M	67.51	75.84	31.50	36.05	24.41	28.10	53.26	60.50	71.23	78.91	49.58	55.88
RelayFormer	2026 ICLR	91.9M	70.29	78.02	35.95	41.54	30.50	35.51	57.91	65.76	45.22	54.78	47.97	55.12
RITA	2026 CVPR	55.4M	66.82	75.46	43.15	51.18	29.12	34.81	65.01	71.92	69.04	76.47	54.63	61.97
FLAME	2026 ICML	75.0M	44.23	53.81	32.14	42.14	27.12	34.65	40.15	51.24	46.18	58.13	37.96	47.99
PANet	-	74.9M	**78.59**	**85.17**	**46.43**	**53.38**	**32.69**	**36.41**	**66.36**	**73.19**	**74.82**	**81.98**	**59.78**	**66.03**

Implementation Details. We use SegFormer-B2 (Xie et al., 2021) as the semantic encoder and ConvNeXt V2-Tiny (Woo et al., 2023) as the forensic encoder. We set the number of Log-Gabor scales to $S = 4$, using normalized center frequencies $\{0.04, 0.08, 0.16, 0.32\}$. We train PANet for 45 epochs using the AdamW optimizer, with an input resolution of 512×512. The learning rate is set to 2×10^{-4} with a weight decay of 5×10^{-3}. All training is conducted on 2 NVIDIA V100 GPUs.

5.2 Comparison with State-Of-The-Art Models

5.2.1 Pixel-Level Cross-Domain Generalization

Table 2 reports the pixel-level localization results, where all methods are trained on our Impostor dataset and then directly evaluated on different datasets, including four out-of-domain benchmarks. As shown, PANet achieves the best overall performance, with an average IoU/F1 of 59.78/66.03, outperforming the strongest baseline by 5.15% in IoU and 4.06% in F1. Notably, PANet ranks first across all five datasets, indicating strong localization performance and cross-domain generalization. We attribute these gains to the explicit modeling of the forensic-semantic mismatch introduced by AIGC editing. The semantic branch uses scene structure to guide localization, while the forensic branch captures residual cues and local phase disruptions in visually plausible manipulated regions. The FSC loss further uses the semantic-forensic mismatch to provide discriminative guidance for localization. Moreover, the strong cross-domain performance of models trained on Impostor suggests that Impostor can serve as an effective and transferable training source for IMDL.

5.2.2 Image-Level Cross-Domain Generalization

In this experiment, models are first trained on ProGAN and then directly evaluated on various out-of-domain datasets without fine-tuning to assess their generalization capacity. Following previous works (Zhong et al., 2024; Yan et al., 2024), each image is divided into 32×32 patches and ranked by DCT-based high-frequency energy. Then, the highest- and lowest-ranked patches are selected as texture-rich and texture-poor inputs, respectively, which are fed into the model for manipulation detection. This patch-based protocol changes only the input construction, while the PANet architecture remains unchanged. For image-level detection, each image is treated as entirely real or generated, and its image-level label is assigned to all spatial locations for FSC and TraceNCE supervision.

Table 3 reports the experimental results. PANet achieves the highest mean accuracy of 94.42%, outperforming the strongest baseline AIDE by 3.17%. PANet ranks first on 13 of the 17 test datasets and second on three other datasets, showing robust generalization across diverse generators. Notably, Impostor is a particularly challenging target domain, on which most baselines show clear performance drops. Nevertheless, PANet still achieves the best accuracy of 71.19%, exceeding the second-best result by 4.18%. Together with the pixel-level results, these findings show that PANet generalizes well across unseen domains at both pixel and image levels.

5.3 Ablation Studies and Robustness of PANet

In Table 4, we provide the ablation results of PANet. For forensic representation ablation, the raw-RGB variant obtains an IoU of 56.04, while LPCE-only and RCE-only improve it to 58.17 and

Table 3: Image-level detection performance under cross-domain evaluation on AIGCDetectBenchmark (Zhong et al., 2024). **Bold** and underline denote the best and second-best results, respectively.

Method	*ProGAN*	*StyleGAN*	*BigGAN*	*CycleGAN*	*StarGAN*	*GauGAN*	*StyleGAN2*	*WFIR*	*ADM*	*Glide*	*Midjourney*	*SD v1.4*	*SD v1.5*	*VQDM*	*Wukong*	*DALLE2*	*Impostor*	*Average*
CNNSpot	**100.00**	90.17	71.17	87.62	94.60	81.42	86.91	91.65	60.39	58.07	51.39	50.57	50.53	56.46	51.03	50.45	50.27	69.57
FreDect	99.36	78.02	81.97	78.77	94.62	80.57	66.19	50.75	63.42	54.13	45.87	38.79	39.21	77.80	40.30	34.70	33.85	62.25
Fusing	**100.00**	85.20	77.40	87.00	97.00	77.00	83.30	66.80	49.00	57.20	52.20	51.00	51.40	55.10	51.70	52.80	49.77	67.29
Gram-Net	99.99	87.05	67.33	86.07	95.05	69.35	87.28	86.80	58.61	54.50	50.02	51.70	52.16	52.86	50.76	49.25	48.12	67.46
LNP	99.95	92.64	88.43	79.07	**100.00**	79.17	93.82	50.00	83.91	83.50	69.55	89.33	88.81	85.03	86.39	92.45	55.74	83.40
UnivFD	99.81	84.93	95.08	98.33	95.75	**99.47**	74.96	86.90	66.87	62.46	56.13	63.66	63.49	85.31	70.93	50.75	50.78	76.80
DIRE-G	95.19	83.03	70.12	74.19	95.47	67.79	75.31	58.05	75.78	71.75	58.01	49.74	49.83	53.68	54.46	66.48	50.94	67.64
PatchCraft	**100.00**	92.77	95.80	70.17	99.97	71.58	89.55	85.80	82.17	83.79	90.12	95.38	95.30	88.91	91.07	96.60	59.27	87.54
NPR	99.79	97.70	84.35	96.10	99.35	82.50	98.38	65.80	69.69	78.36	77.85	78.63	78.89	78.13	76.11	64.90	53.18	81.16
AIDE	99.99	99.64	83.95	98.48	99.91	73.25	98.00	**94.20**	93.43	**95.09**	77.20	93.00	92.85	95.16	93.55	96.60	67.01	91.25
PANet	99.99	**99.74**	**96.53**	**98.71**	**100.00**	93.52	**98.86**	73.15	**94.48**	93.37	**93.29**	**98.96**	**98.99**	**97.46**	**98.67**	**98.25**	**71.19**	**94.42**

Table 4: Ablation study of PANet. We report the average IoU and F1 over all five localization benchmarks.

Variant	Average IoU	Average F1
Level I: Forensic Representation Ablation		
w/o RCE (LPCE-only)	58.17	63.95
w/o LPCE (RCE-only)	58.34	64.04
w/o LFCE (Raw RGB)	56.04	61.60
Level II: Framework Component Ablation		
w/o Forensic Branch	54.43	60.04
w/o MGCA	57.41	63.59
w/o FSC Loss	58.37	64.02
w/o TraceNCE Loss	59.20	64.91
PANet	**59.78**	**66.03**

Table 5: Robustness analysis under common post-processing distortions, measured by Average IoU.

Distortion	NFA-ViT	M2SFormer	Mesorch	RITA	PANet
no distortion	49.58	43.90	53.79	54.63	**59.78**
Resize(0.78×)	44.35	41.85	53.11	53.44	**58.25**
Resize(0.25×)	31.52	33.48	36.33	41.68	**46.05**
Blur($k = 3$)	43.75	44.91	50.50	53.37	**57.56**
Blur($k = 7$)	36.10	39.85	39.33	43.72	**49.28**
Noise($\sigma = 3$)	20.57	30.36	39.07	42.38	**44.30**
Noise($\sigma = 5$)	12.34	21.87	32.77	37.67	**38.77**
Compress($q = 95$)	28.28	38.18	40.72	39.61	**48.78**
Compress($q = 90$)	25.28	36.48	36.02	35.94	**45.78**
Compress($q = 85$)	20.12	34.57	30.71	33.71	**39.34**

58.34, respectively. Notably, LPCE alone yields a 2.13% gain over raw RGB. Combining both cues further increases IoU to 59.78, showing their effectiveness and complementarity. For framework-level components, removing the forensic branch causes the largest performance drop, showing that semantic features alone are insufficient for robust localization. Removing the FSC loss decreases IoU from 59.78 to 58.37, showing that forensic-semantic consistency supervision provides effective localization guidance. MGCA and TraceNCE also provide consistent improvements.

Moreover, the performance of IMDL methods can be affected by image degradations during transmission and storage. To evaluate robustness under such conditions, we simulate common degradations introduced by post-processing, including resizing, Gaussian blur, Gaussian noise, and JPEG compression, with different distortion strengths. As shown in Table 5, although all methods experience performance degradation under stronger distortions, PANet consistently achieves the highest Average IoU across all settings.

6 Conclusion

In this paper, we present **Impostor**, a high-quality benchmark for IMDL on realistic AI-edited images. Impostor is curated by **DisAgent**, a closed-loop agent framework that automatically generates realistic AI-edited images and corresponding pixel-level masks. Compared with existing benchmarks, Impostor provides higher visual realism, greater manipulation diversity, and broader generator coverage, better reflecting localized editing in real-world scenarios. To address the challenges posed by these realistic manipulations, we propose **PANet**, which combines local phase modeling and forensic-semantic consistency learning. PANet achieves state-of-the-art manipulation localization performance and strong cross-dataset generalization. We hope that DisAgent can support the continual expansion of Impostor with emerging editing models, allowing the benchmark to keep pace with generative editing and promote long-term progress in IMDL.

AI USE STATEMENT

Generative AI tools were used in the construction of the Impostor dataset, including LVLM/LLM-based planning, validation, and reflection, as well as generative image editing. LVLMs were also used for the visual-realism assessment described in the paper. Limited generative AI assistance was used for manuscript language editing. The PANet methodology, mathematical formulation, and core implementation were developed by the authors. All AI-assisted outputs were reviewed and verified by the authors, who take full responsibility for the final content of this work.

ETHICS STATEMENT

Impostor is constructed from images in LVIS for research on AI-generated image manipulation detection and localization. We plan to release the manipulated images, manipulation masks, editing prompts, generation metadata, and generation code to support reproducibility and benchmarking. Because these resources can facilitate the generation of realistic image manipulations, we acknowledge their potential misuse for creating deceptive visual content. The human verification stage is limited to assessing image naturalness and manipulation-mask alignment. The release of the dataset and associated resources will follow the applicable licenses and terms of the source data.

REFERENCES

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

Lvpan Cai, Haowei Wang, Jiayi Ji, YanShu ZhouMen, Shen Chen, Taiping Yao, and Xiaoshuai Sun. Zooming in on fakes: A novel dataset for localized ai-generated image detection with forgery amplification approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 2534–2542, 2026.

Lucy Chai, David Bau, Ser-Nam Lim, and Phillip Isola. What makes fake images detectable? understanding properties that generalize. In *European conference on computer vision*, pp. 103–120. Springer, 2020.

Yirui Chen, Xudong Huang, Quan Zhang, Wei Li, Mingjian Zhu, Qiangyu Yan, Simiao Li, Hanting Chen, Hailin Hu, Jie Yang, et al. Gim: A million-scale benchmark for generative image manipulation detection and localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 2311–2319, 2025.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Chengbo Dong, Xinru Chen, Ruohan Hu, Juan Cao, and Xirong Li. 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.

Jing Dong, Wei Wang, and Tieniu Tan. Casia image tampering detection evaluation database. In *2013 IEEE China summit and international conference on signal and information processing*, pp. 422–426. IEEE, 2013.

Michael Felsberg and Gerald Sommer. The monogenic signal. *IEEE transactions on signal processing*, 49(12):3136–3144, 2001.

Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. In *International conference on machine learning*, pp. 3247–3258. PMLR, 2020.

Jessica Fridrich and Jan Kodovsky. Rich models for steganalysis of digital images. *IEEE Transactions on information Forensics and Security*, 7(3):868–882, 2012.

Google DeepMind. Gemini 3 pro image model card. Technical report, November 2025. URL `https://storage.googleapis.com/deepmind-media/Model-Cards/Gemini-3-Pro-Image-Model-Card.pdf`.

Google DeepMind. Gemini 3 pro model card. Technical report, November 2025b. URL `https://storage.googleapis.com/deepmind-media/Model-Cards/Gemini-3-Pro-Model-Card.pdf`.

Fabrizio Guillaro, Davide Cozzolino, Avneesh Sud, Nicholas Dufour, and Luisa 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*, pp. 20606–20615, 2023.

Xiao Guo, Xiaohong Liu, Zhiyuan Ren, Steven Grosz, Iacopo Masi, and Xiaoming Liu. 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.

Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5356–5364, 2019.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Wen Huang, Jiarui Yang, Tao Dai, Jiawei Li, Shaoxiong Zhan, Bin Wang, and Shu-Tao Xia. Relayformer: A unified local-global attention framework for scalable image and video manipulation localization. In *International Conference on Learning Representations*, volume 2026, pp. 89421–89443, 2026.

Zhenglin Huang, Jinwei Hu, Xiangtai Li, Yiwei He, Xingyu Zhao, Bei Peng, Baoyuan Wu, Xiaowei Huang, and Guangliang Cheng. Sida: Social media image deepfake detection, localization and explanation with large multimodal model. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 28831–28841, 2025.

Shan Jia, Mingzhen Huang, Zhou Zhou, Yan Ju, Jialing Cai, and Siwei Lyu. 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.

Yan Ju, Shan Jia, Lipeng Ke, Hongfei Xue, Koki Nagano, and Siwei Lyu. Fusing global and local features for generalized ai-synthesized image detection. In *2022 IEEE International Conference on Image Processing (ICIP)*, pp. 3465–3469. IEEE, 2022.

Myung-Joon Kwon, Seung-Hun Nam, In-Jae Yu, Heung-Kyu Lee, and Changick Kim. Learning jpeg compression artifacts for image manipulation detection and localization. *International Journal of Computer Vision*, 130(8):1875–1895, aug 2022. doi: 10.1007/s11263-022-01617-5.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025.

Dong Li, Jiaying Zhu, Yidi Liu, Xin Lu, Xueyang Fu, Jiawei Liu, Aiping Liu, and Zheng-Jun Zha. Learnable frequency decomposition for image forgery detection and localization. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 1359–1367, 2025.

Bo Liu, Fan Yang, Xiuli Bi, Bin Xiao, Weisheng Li, and Xinbo Gao. Detecting generated images by real images. In *European conference on computer vision*, pp. 95–110. Springer, 2022a.

Shiyu Liu, Yucheng Han, Peng Xing, Fukun Yin, Rui Wang, Wei Cheng, Jiaqi Liao, Yingming Wang, Honghao Fu, Chunrui Han, et al. Step1x-edit: A practical framework for general image editing. *arXiv preprint arXiv:2504.17761*, 2025.

Xiaohong Liu, Yaojie Liu, Jun Chen, and Xiaoming Liu. 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, 2022b.

Zhengzhe Liu, Xiaojuan Qi, and Philip HS Torr. Global texture enhancement for fake face detection in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8060–8069, 2020.

Xiaochen Ma, Bo Du, Zhuohang Jiang, Ahmed Y. Al Hammadi, and Jizhe Zhou. Iml-vit: Benchmarking image manipulation localization by vision transformer, 2023.

Ju-Hyeon Nam, Dong-Hyun Moon, and Sang-Chul Lee. M2sformer: Multi-spectral and multi-scale attention with edge-aware difficulty guidance for image forgery localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15927–15938, 2025.

Tian-Tsong Ng, Shih-Fu Chang, and Q Sun. A data set of authentic and spliced image blocks. *Columbia University, ADVENT Technical Report*, 4:2033–2004, 2004.

Adam Novozamsky, Babak Mahdian, and Stanislav Saic. Imd2020: A large-scale annotated dataset tailored for detecting manipulated images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision workshops*, pp. 71–80, 2020.

Utkarsh Ojha, Yuheng Li, and Yong Jae Lee. Towards universal fake image detectors that generalize across generative models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 24480–24489, 2023.

OpenAI. Gpt-5.1 instant and gpt-5.1 thinking system card addendum. Technical report, November 2025. URL https://cdn.openai.com/pdf/4173ec8d-1229-47db-96de-06d87147e07e/5_1_system_card.pdf.

OpenAI. Gpt-5.4 thinking system card. Technical report, March 2026. URL https://openai.com/index/gpt-5-4-thinking-system-card/.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. In *International Conference on Learning Representations*, volume 2025, pp. 28085–28128, 2025.

Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024.

Chuangchuang Tan, Yao Zhao, Shikui Wei, Guanghua Gu, Ping Liu, and Yunchao Wei. Rethinking the up-sampling operations in cnn-based generative network for generalizable deepfake detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 28130–28139, 2024.

Junke Wang, Zuxuan Wu, Jingjing Chen, Xintong Han, Abhinav Shrivastava, Ser-Nam Lim, and Yu-Gang Jiang. Objectformer for image manipulation detection and localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2364–2373, 2022.

Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. Cnn-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8695–8704, 2020.

Yabin Wang, Zhiwu Huang, and Xiaopeng Hong. Opensdi: Spotting diffusion-generated images in the open world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4291–4301, 2025.

Yiming Wang, Baiqi Wu, Qingming Li, Jiahao Chen, Tong Zhang, and Shouling Ji. Order within chaos: Capturing intrinsic energy anomalies for ai-manipulated image forgery localization. *arXiv preprint arXiv:2606.02178*, 2026.

Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, Hezhen Hu, Hong Chen, and Houqiang Li. Dire for diffusion-generated image detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22445–22455, 2023.

Runpu Wei, Zijin Yin, Shuo Zhang, Lanxiang Zhou, Xueyi Wang, Chao Ban, Tianwei Cao, Hao Sun, Zhongjiang He, Kongming Liang, and Zhanyu Ma. Omnieraser: Remove objects and their effects in images with paired video-frame data. *arXiv preprint arXiv:2501.07397*, 2025. URL https://arxiv.org/abs/2501.07397.

Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie. Convnext v2: Co-designing and scaling convnets with masked autoencoders. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16133–16142, 2023.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025.

Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34:12077–12090, 2021.

Zhipei Xu, Xuanyu Zhang, Runyi Li, Zecheng Tang, Qing Huang, and Jian Zhang. Fakeshield: Explainable image forgery detection and localization via multi-modal large language models. In *International Conference on Learning Representations*, 2025.

Shilin Yan, Ouxiang Li, Jiayin Cai, Yanbin Hao, Xiaolong Jiang, Yao Hu, and Weidi Xie. A sanity check for ai-generated image detection. *arXiv preprint arXiv:2406.19435*, 2024.

Sidi Yang, Tianhe Wu, Shuwei Shi, Shanshan Lao, Yuan Gong, Mingdeng Cao, Jiahao Wang, and Yujiu Yang. Maniqa: Multi-dimension attention network for no-reference image quality assessment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1191–1200, 2022.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*, 2022.

Nan Zhong, Yiran Xu, Sheng Li, Zhenxing Qian, and Xinpeng Zhang. Patchcraft: Exploring texture patch for efficient ai-generated image detection. 2024. URL https://arxiv.org/abs/2311.12397.

Xuekang Zhu, Xiaochen Ma, Lei Su, Zhuohang Jiang, Bo Du, Xiwen Wang, Zeyu Lei, Wentao Feng, Chi-Man Pun, and Ji-Zhe Zhou. Mesoscopic insights: orchestrating multi-scale & hybrid architecture for image manipulation localization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 11022–11030, 2025.

Xuekang Zhu, Ji-Zhe Zhou, Kaiwen Feng, Chenfan Qu, Xiwen Wang, Yunfei Wang, Liting Zhou, and Jian Liu. Revisiting image manipulation localization under realistic manipulation scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7198–7207, 2026.

Junhao Zhuang, Yanhong Zeng, Wenran Liu, Chun Yuan, and Kai Chen. A task is worth one word: Learning with task prompts for high-quality versatile image inpainting. In *European Conference on Computer Vision*, pp. 195–211. Springer, 2024.

APPENDIX

A FURTHER ANALYSIS OF PANET DESIGN

A.1 EMPIRICAL SUPPORT FOR THE FORENSIC-SEMANTIC CONSISTENCY (FSC) PRIOR

Setup and Metric. We conduct a diagnostic experiment to test the hypothesis underlying FSC: the forensic-semantic correspondence learned from authentic images is better preserved in authentic regions than in AIGC-manipulated regions. We first train a semantic-to-forensic projector using 1,000 authentic images. For each image, a frozen DINOv2 encoder extracts high-level semantic features from the RGB image, while a frozen ConvNeXt encoder extracts low-level forensic features from the forensic cues generated by the LFCE module. Let s_i and f_i denote the semantic and forensic features at spatial position i, respectively. We train a lightweight MLP projector $g_\theta(\cdot)$ to predict the forensic feature from its corresponding semantic feature using an MSE loss:

$$\hat{f}_i = g_\theta(s_i), \quad \mathcal{L}_{\text{proj}} = \text{MSE}(\hat{f}_i, f_i), \tag{9}$$

where $\hat{f}_i$ denotes the predicted forensic feature. After training, we freeze g_θ and evaluate it on manipulated images. Specifically, we randomly sample 200 manipulated images from each of the ten subsets of Impostor, resulting in 2,000 test images covering different manipulation types and editing models. The 1,000 authentic images used for projector training do not overlap with the 2,000 manipulated images used for testing. Let Ω_{real} and Ω_{fake} denote the authentic and manipulated regions defined by the ground-truth mask, respectively. At each spatial position i, we compute the local FSC score as

$$\text{FSC}_i = 100 \cdot \text{CosSim}(\hat{f}_i, f_i) \tag{10}$$

Here, $\text{CosSim}(\cdot, \cdot)$ denotes cosine similarity, and the factor of 100 is used for easier interpretation. A higher FSC_i indicates stronger agreement between the forensic feature predicted and the actual forensic feature, corresponding to stronger forensic-semantic consistency.

For each manipulated image, we then compute the average FSC scores over the authentic and manipulated regions and define their difference as the FSC gap:

$$\text{FSC}_{\text{real}} = \frac{1}{|\Omega_{\text{real}}|} \sum_{i \in \Omega_{\text{real}}} \text{FSC}_i, \ \text{FSC}_{\text{fake}} = \frac{1}{|\Omega_{\text{fake}}|} \sum_{i \in \Omega_{\text{fake}}} \text{FSC}_i, \ \Delta_{\text{FSC}} = \text{FSC}_{\text{real}} - \text{FSC}_{\text{fake}}. \tag{11}$$

Here, Δ_{FSC} measures the difference in forensic-semantic consistency between authentic and manipulated regions. A positive Δ_{FSC} indicates that forensic-semantic consistency is higher in authentic regions than in manipulated regions, while a larger Δ_{FSC} indicates a greater relative difference between the two regions. We use Δ_{FSC} as the primary measure in the following analysis.

Results and controls. Figure 4 and Table 6 provide empirical support for the FSC prior. Under the RGB-Forensic setting, authentic regions achieve clearly higher FSC scores than manipulated regions, with an average gap of $+16.10$. Moreover, $\Delta_{\text{FSC}} > 0$ holds for 89.5% of the test images, showing that this pattern occurs in the majority of samples. To further examine whether the observed FSC gap could arise from pre-existing content differences between the target region and its surroundings, we introduce a source-image control. For each manipulated image, we apply the same manipulation mask to its corresponding unedited source image and compute the FSC gap over the same spatial regions. The source control yields a much smaller gap of $+3.98$, compared with $+16.10$ after manipulation, indicating that AIGC editing disrupts low-level forensic traces in the edited regions, thereby weakening their correspondence with semantic representations. When ground-truth masks are randomly shuffled across images, the gap nearly disappears (-0.17), indicating that the observed difference is specifically associated with the actual manipulated regions rather than arbitrary spatial partitions. The FFT low-frequency controls produce much smaller gaps ($+1.37$ and $+2.85$), indicating that the correspondence between semantic content and generic low-frequency information remains largely preserved in manipulated regions. The substantially larger RGB-Forensic gap therefore highlights the stronger sensitivity of forensic cues to localized AIGC editing. Together, these controls provide empirical support for the FSC prior.

Figure 5 further reports the FSC scores for each subset under the RGB-Forensic setting. All ten subsets show positive FSC gaps, indicating that authentic regions consistently exhibit higher FSC scores than manipulated regions across different editing models and manipulation types.

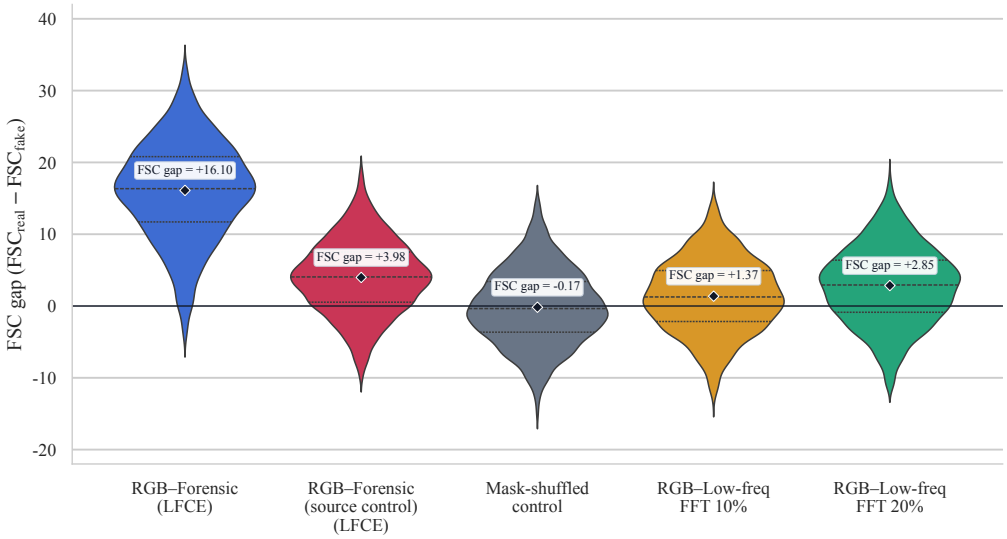

Figure 4: Distribution of image-level FSC gaps under the RGB-Forensic setting and different control settings. For the source control, each manipulation mask is applied to its corresponding unedited source image, and the FSC gap is computed between the mask-outside and mask-inside regions at the same spatial locations. Since both regions are authentic in the source image, this control measures pre-existing region-dependent differences unrelated to AIGC editing. Mask-shuffle randomly permutes the ground-truth masks across test images to test whether the FSC gap is associated with the actual manipulated regions rather than arbitrary spatial partitions. Low-frequency controls replace LFCE-based forensic cues with FFT low-frequency cues to examine whether the FSC gap can still be observed without forensic cues

Table 6: Quantitative comparison of FSC scores under different settings. The $\Delta_{\mathrm{FSC}} > 0$ (%) column reports the percentage of test images with a positive FSC gap. The control settings follow the definitions in Figure 4

Setting	Images	Image Type	$\mathrm{FSC}_{\mathrm{real}}$	$\mathrm{FSC}_{\mathrm{fake}}$	Δ_{FSC}	$\Delta_{\mathrm{FSC}} > 0$ (%)
RGB-Forensic main	2000	manipulated	62.05	45.96	+16.10	89.5%
RGB-Forensic source control	2000	Authentic	63.23	59.24	+3.98	69.3%
Mask-shuffle control	2000	manipulated	61.62	61.79	−0.17	44.3%
RGB-Low-freq 10% control	2000	manipulated	82.76	81.39	+1.37	55.3%
RGB-Low-freq 20% control	2000	manipulated	78.87	76.02	+2.85	60.4%

Implication for FSC. These diagnostic results support the motivation of FSC: authentic regions generally exhibit stronger forensic–semantic consistency, whereas AIGC editing tends to disrupt such consistency. Since this gap is already observable with frozen feature encoders, it motivates explicitly supervising the FSC difference between authentic and manipulated regions in PANet. Notably, $\mathrm{FSC}_{\mathrm{real}} > \mathrm{FSC}_{\mathrm{fake}}$holds for 89.5% of the test images, indicating a strong but not absolute tendency. This observation is consistent with our FSC design: FSC encourages authentic regions to produce stronger FSC responses and enlarges the relative gap between authentic and manipulated regions, without forcing every manipulated region to have low consistency.

A.2 Empirical Support for Local Phase Modeling

To empirically examine the motivation behind LPCE, we investigate whether local phase representations can capture forensic cues that are sensitive to localized AIGC manipulation. Specifically, we analyze the LPCE phase responses before the learnable convolution and further evaluate their ability to distinguish authentic and manipulated regions.

As shown in Figure 6, the LPCE responses preserve spatially aligned structural patterns, including edges, contours, and local textures. Localized AIGC editing can disturb these cues, causing manipulated regions to exhibit different phase responses from authentic regions. To further quantify this observation, we conduct a patch-level separability analysis and compare LPCE with conventional

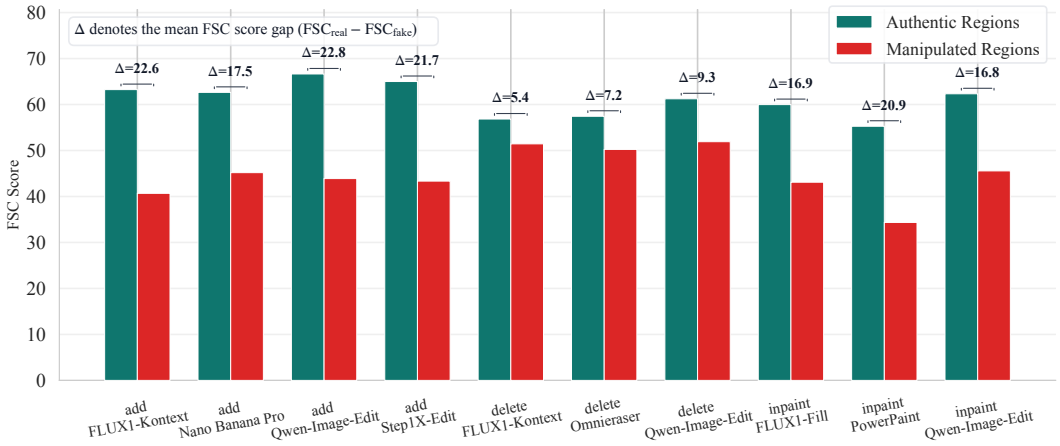

Figure 5: Per-subset comparison of FSC scores between authentic and manipulated regions. Authentic regions exhibit higher FSC scores across all ten subsets.

frequency-domain representations based on DCT, DWT, and FFT under the same evaluation protocol. Following common practice in image forensics(Guillaro et al., 2023; Wang et al., 2022; Li et al., 2025), we use the high-frequency components of DCT, DWT, and FFT as conventional frequency-domain baselines, since they emphasize local texture and edge variations widely used for detecting manipulation traces.

Given an LPCE phase response map or the response map of a frequency-domain baseline, together with the corresponding manipulation mask, we divide each response map into non-overlapping 32×32 patches. Patches containing both authentic and manipulated pixels are excluded to reduce label ambiguity, leaving only purely authentic or manipulated patches. For each retained patch p, we extract a fixed set of four generic summary statistics from its response values:

$$\mathbf{z}_p = [\mu_p, \sigma_p, H_p, \mathrm{IQR}_p]. \tag{12}$$

Let $\{x_{p,j}\}_{j=1}^{N}$ denote the N response values within patch p. The mean and standard deviation are computed as

$$\mu_p = \frac{1}{N}\sum_{j=1}^{N} x_{p,j}, \quad \sigma_p = \sqrt{\frac{1}{N}\sum_{j=1}^{N}(x_{p,j} - \mu_p)^2}. \tag{13}$$

The histogram entropy is defined as

$$H_p = -\sum_{b=1}^{16} q_{p,b} \log(q_{p,b} + \epsilon), \tag{14}$$

where $q_{p,b}$ denotes the normalized frequency of the b-th histogram bin, and ϵ is a small constant for numerical stability. Finally, $\mathrm{IQR}_p = Q_{75,p} - Q_{25,p}$ denotes the interquartile range of the patch responses.

To quantify how much each patch deviates from authentic-region statistics, we estimate reference statistics from a subset of authentic patches. Let $\boldsymbol{\mu}_a$ and $\boldsymbol{\Sigma}_a$ denote the mean vector and covariance matrix of their descriptors, respectively. For each evaluated patch descriptor $\mathbf{z}_p$, we compute its regularized Mahalanobis distance:

$$d(\mathbf{z}_p) = \sqrt{(\mathbf{z}_p - \boldsymbol{\mu}_a)^\top (\boldsymbol{\Sigma}_a + \lambda \mathbf{I})^{-1} (\mathbf{z}_p - \boldsymbol{\mu}_a)}, \tag{15}$$

where λ is a small regularization coefficient added to the covariance matrix to ensure stable inversion, and $\mathbf{I}$ denotes the identity matrix. A larger $d(\mathbf{z}_p)$ indicates a stronger statistical deviation from the authentic reference statistics. To avoid biased comparison, the authentic patches used to estimate the reference statistics are disjoint from those used for evaluation.

Figure 7 shows the distributions of the deviation scores for authentic and manipulated patches. LPCE produces a clearer separation between the two distributions than the DCT-, DWT-, and

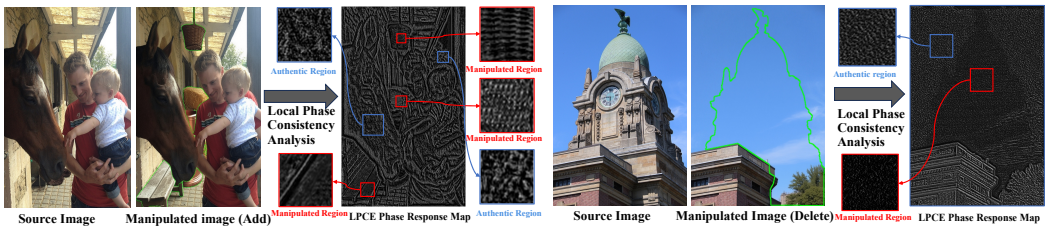

Figure 6: Qualitative visualization of local phase responses. These responses highlight spatially aligned structural traces such as edges, contours, and local textures. Localized AIGC editing can perturb these traces, leading manipulated regions to exhibit response patterns that differ from those in authentic regions.

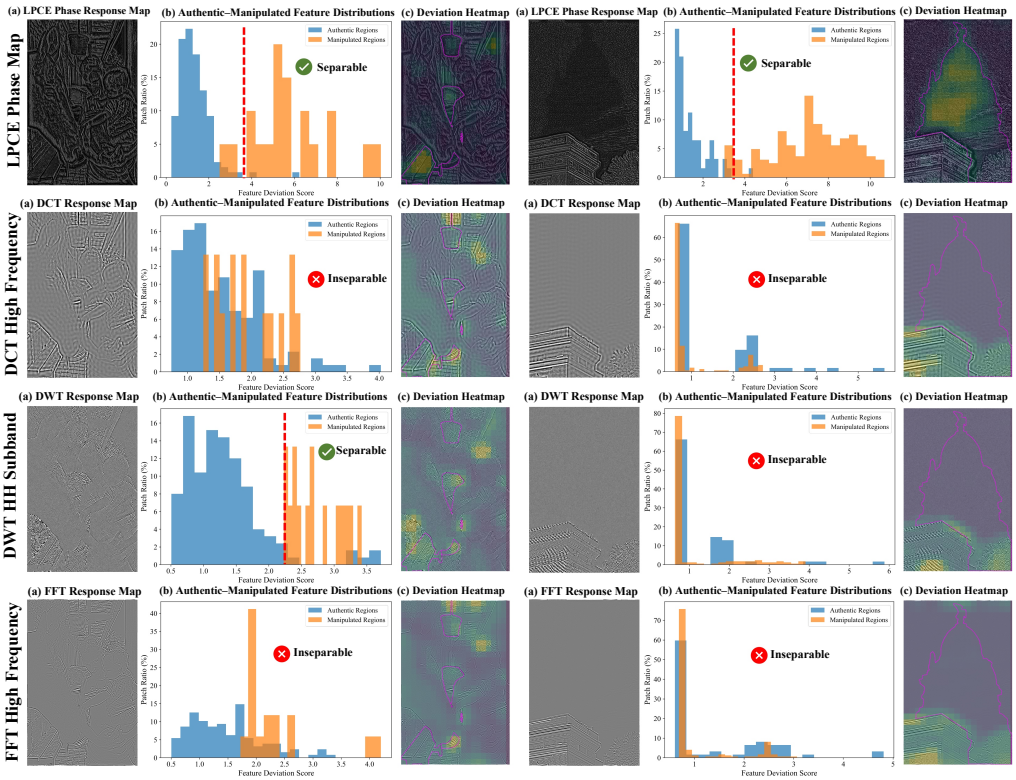

Figure 7: Patch-level statistical separability comparison between LPCE phase responses and conventional frequency-domain baselines. The left and right panels show representative examples of addition-type and deletion-type manipulations, respectively.

FFT-based representations. Authentic patches remain close to the reference statistics, whereas manipulated patches generally exhibit larger deviations. In comparison, conventional frequency-domain representations show considerably more overlap between authentic and manipulated patches. This difference is particularly clear in the deletion example, where the manipulated region is filled with visually smooth background content and exhibits relatively weak high-frequency differences. Under this condition, DCT-, DWT-, and FFT-based representations provide limited separation. In contrast, LPCE captures spatially aligned structural patterns associated with edges, contours, and local textures, making it more sensitive to structural disruptions even when high-frequency differences are weak. Consequently, manipulated patches exhibit larger deviations from authentic reference statistics under LPCE, while the corresponding deviation heatmaps show stronger responses within the manipulated regions. These results support the use of local phase modeling for capturing manipulation-sensitive structural cues that are less evident in conventional frequency-domain representations.

Table 7: Pixel-level generalization with different AIGC training sources. Entries marked with * correspond to the training dataset and are excluded when computing the overall generalization mean.

Method	Train Dataset	SID-Set		BR-Gen		CocoGlide		AutoSplice		Impostor		Average	
		IoU	F1	IoU	F1	IoU	F1	IoU	F1	IoU	F1	IoU	F1
M2SFormer	Impostor	33.97	40.64	24.38	30.76	53.23	61.82	54.41	65.27	*53.50	*64.24	41.50	49.62
M2SFormer	BR-Gen	27.24	32.81	*85.15	*90.52	51.82	59.67	52.03	59.27	15.48	20.05	36.64	42.95
M2SFormer	SID-Set	*68.64	*77.53	12.51	16.75	53.72	59.52	30.19	36.92	13.17	16.06	27.40	32.31
Mesorch	Impostor	39.08	44.46	24.57	28.06	64.23	71.61	70.30	77.58	*70.79	*78.76	49.55	55.43
Mesorch	BR-Gen	37.51	41.03	*92.26	*95.30	62.14	68.15	64.02	69.91	11.78	14.06	43.86	48.29
Mesorch	SID-Set	*87.04	*91.83	26.52	28.75	56.72	62.81	20.65	23.92	20.18	24.23	31.02	34.93
PANet	Impostor	46.43	53.38	32.69	36.41	66.36	73.19	74.82	81.98	*78.59	*85.17	55.08	61.24
PANet	BR-Gen	40.20	46.25	*92.68	*95.44	63.54	70.32	70.56	75.63	17.78	21.14	48.02	53.34
PANet	SID-Set	*87.72	*92.04	28.31	33.14	62.14	65.18	36.25	40.71	16.27	19.52	35.74	39.64

B Additional Quantitative Results

B.1 Generalization Analysis with Different Training Sources.

To further validate the generalization value of Impostor as a training source, we conduct cross-dataset generalization experiments with different training sources. As shown in Table 7, we select two competing localization models with strong generalization capability, together with PANet. Each model is trained separately on Impostor, BR-Gen (Cai et al., 2026), and SID-Set (Huang et al., 2025), and then evaluated on the remaining AIGC manipulation localization benchmarks. To avoid in-domain bias, we exclude performance on the corresponding training dataset when computing the average score, enabling a fairer comparison across different training sources.

The results show that, across all three models, training on Impostor consistently yields substantially better average cross-dataset performance than training on BR-Gen or SID-Set. This consistent advantage indicates that the generalization benefit of Impostor is not limited to a particular model, demonstrating its value as a training source for AIGC manipulation localization.

We attribute this advantage to the diversity and realism of Impostor. As illustrated in Figure 2, Impostor covers three major manipulation types and further includes multi-region manipulation scenarios, exposing models to diverse localization patterns during training. It also incorporates multiple recent image editing models, providing broader generator coverage and more diverse manipulation traces. In addition, the agent-based closed-loop construction pipeline improves editing-intent plausibility, contextual consistency, and visual realism. Together, these properties provide diverse and challenging training samples, enabling models to learn more transferable manipulation cues and generalize better to unseen datasets.

B.2 Model Universality Across AIGC Localization Benchmarks.

Beyond evaluating the effectiveness of Impostor as a training source, we further investigate whether the proposed model consistently performs well across different AIGC manipulation localization benchmarks. To this end, we conduct in-domain evaluations on four datasets, including BR-Gen, SID-Set, OpenSDI, and Impostor. For each benchmark, all compared methods are independently trained and evaluated on the same dataset under a unified protocol, with the training/test splits and evaluation settings following Sec. 3.4. We additionally report the average performance across all four datasets to provide an overall assessment of each method's universality across diverse AIGC manipulation localization benchmarks.

As shown in Table 8 and Figure 8, PANet achieves the best average performance across the four benchmarks and ranks first on all four datasets. Note that these in-domain results are not directly comparable to the cross-domain results in Table 7, as strong in-domain performance does not necessarily imply better cross-domain generalization. These results demonstrate that the effectiveness of our model is not confined to a specific dataset, but remains consistent across diverse localized AIGC manipulation benchmarks. We attribute this advantage to local phase modeling and forensic-semantic consistency learning. The former captures manipulation-sensitive local forensic cues, while the latter

explicitly models the consistency between semantic and forensic representations, enabling more effective manipulation localization across different data distributions.

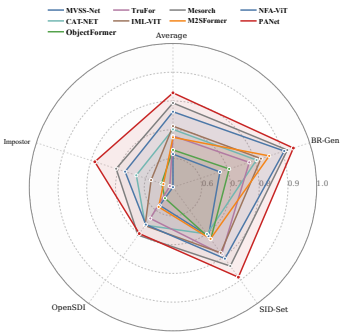

Figure 8: Radar Chart of Pixel-Level Localization performance across multiple AIGC datasets.

Method	BR-Gen		SID-Set		OpenSDI		Impostor		Average	
	IoU	F1	IoU	F1	IoU	F1	IoU	F1	IoU	F1
MVSS-Net	67.18	74.15	71.02	79.36	57.84	65.31	53.24	61.31	62.32	70.03
CAT-NET	80.57	86.53	70.14	78.11	66.36	74.89	65.14	74.12	70.55	78.41
ObjectFormer	70.57	79.43	72.37	80.54	54.74	64.24	56.13	66.12	63.45	72.58
TruFor	77.91	83.84	78.82	85.67	63.45	71.06	52.24	60.12	68.11	75.17
IML-VIT	82.18	87.12	78.23	83.55	66.54	73.67	58.72	67.91	71.42	78.06
Mesorch	<u>92.26</u>	<u>95.30</u>	<u>86.34</u>	<u>91.94</u>	<u>71.83</u>	<u>78.70</u>	<u>72.81</u>	<u>80.11</u>	<u>80.81</u>	<u>86.51</u>
M2SFormer	86.22	91.53	72.23	80.75	58.49	67.31	55.34	65.91	68.07	76.38
NFA-ViT	90.71	93.22	80.74	86.25	66.08	73.46	69.14	76.96	76.67	82.47
PANet	**94.01**	**96.11**	**90.11**	**94.32**	**73.50**	**81.01**	**80.12**	**87.21**	**84.44**	**89.66**

Table 8: In-domain pixel-Level localization performance across different AIGC Benchmarks. **Bold** and <u>underline</u> denote the best and second-best results, respectively.

B.3 Pixel-Level In-Domain Analysis on Impostor

To further investigate how manipulation types and editing models affect localization performance, we conduct a fine-grained in-domain analysis on **Impostor**. Unlike the cross-dataset evaluation in Sec. 5.2.1, this analysis examines how localization difficulty varies within Impostor across different manipulation types and editing models. Specifically, we partition the Impostor test set into 10 subsets according to the manipulation type and editing model, covering three major categories: Add, Delete, and Inpaint. The results are presented in Table 9. Since these subsets contain different numbers of samples, the subset-wise average may differ slightly from the overall result in Table 2, which is computed over the entire test set.

From the perspective of manipulation type, Add is generally the easiest to localize, with Nano-Banana being a notable exception, while Delete is the most challenging and Inpaint exhibits intermediate difficulty. These differences can be attributed to how each manipulation type alters the original image content. Add introduces new content into an existing image and can therefore leave relatively noticeable texture discontinuities, structural mismatches, or semantic inconsistencies. Inpaint edits existing regions or objects under stronger contextual constraints, making such anomalies less pronounced. Delete is typically more challenging because the removed content is replaced by synthesized background that blends naturally with the surrounding regions, leaving fewer distinctive cues for localization. **From the perspective of editing models,** localization performance also varies considerably among models within the same manipulation type, suggesting that localization difficulty is determined not only by the manipulation operation but also by the characteristics of the editing model. In particular, Nano-Banana is substantially more challenging than the other editing models under Add, while clear model-dependent variations are also observed for Delete and Inpaint.

Despite these substantial variations across manipulation types and editing models, PANet achieves the best IoU and F1 scores on all 10 subsets, as well as the highest subset-wise and global averages. These results demonstrate that PANet maintains strong localization performance across diverse manipulation operations and editing models, further supporting the robustness of its local phase modeling and semantic–forensic consistency learning.

B.4 Image-Level In-Domain Analysis on Impostor

To further investigate how different editing models affect image-level detection performance, we conduct a fine-grained in-domain analysis on Impostor, following the same partition protocol as Sec. B.3. The key difference from the localization setting is that each test subset contains equal numbers of positive and negative samples.

Compared with pixel-level localization, image-level detection only requires determining whether an image has been manipulated, without identifying the manipulated region. Therefore, the in-domain detection task is generally easier, and most methods achieve relatively high performance. As shown

Table 9: In-domain pixel-level localization performance on Impostor. For each method, IoU and F1 are reported in the upper and lower rows, respectively. **Bold** and underline denote the best and second-best results, respectively.

Method	Metrics	Add				Delete			Inpaint			Average
		Qwen-Image-Edit	FLUX-Kontext	Step-Edit	Nano-Banana	Qwen-Image-Edit	Omni Eraser	FLUX-Kontext	Qwen-Image-Edit	FLUX-Fill	Power-Paint	
MVSS-Net	IoU	69.58	73.06	61.72	32.04	56.90	19.27	44.72	49.28	55.24	53.33	51.51
	F1	79.78	82.46	72.41	41.93	66.50	25.85	54.41	59.57	64.92	60.13	60.80
CAT-NET	IoU	71.37	76.70	68.08	44.15	55.56	31.84	48.19	64.53	66.61	67.16	59.42
	F1	81.09	84.82	78.09	56.09	65.71	40.34	57.95	73.92	76.10	75.36	68.95
Objectformer	IoU	56.18	61.92	49.99	24.91	50.62	31.93	42.77	61.90	66.04	62.01	50.83
	F1	69.26	74.15	63.05	36.56	62.01	42.38	54.54	72.66	76.84	72.24	62.37
TruFor	IoU	59.40	67.66	52.66	23.90	50.27	22.83	36.85	60.50	59.46	54.93	48.85
	F1	70.11	77.26	63.21	32.20	59.79	28.89	45.45	68.71	68.49	61.38	57.55
IML-VIT	IoU	76.03	81.87	71.14	38.51	57.87	29.74	49.41	58.74	60.90	68.55	59.28
	F1	84.59	88.55	80.46	50.32	67.30	38.84	59.74	68.63	71.03	76.41	68.59
Mesorch	IoU	80.75	83.59	76.31	51.62	68.38	52.07	62.43	77.10	76.82	75.84	70.49
	F1	87.83	89.91	84.30	62.86	76.83	60.91	71.41	84.13	84.28	82.13	78.46
M2SFormer	IoU	68.64	68.00	62.31	36.18	50.76	37.15	45.24	57.92	56.34	58.76	54.13
	F1	79.26	78.82	73.83	49.03	61.48	47.73	56.13	68.20	66.75	68.25	64.95
NFA-ViT	IoU	75.38	80.10	69.05	40.62	63.95	57.29	43.29	73.72	77.08	75.02	65.55
	F1	83.95	87.50	78.24	51.14	72.96	66.72	51.35	81.37	84.51	81.76	73.95
PANet	IoU	**85.26**	**87.99**	**82.91**	**57.31**	**74.52**	**60.83**	**68.63**	**81.96**	**84.93**	**85.25**	**76.96**
	F1	**90.70**	**92.64**	**89.26**	**68.01**	**82.24**	**69.96**	**77.84**	**88.28**	**91.35**	**89.93**	**84.02**

Table 10: In-Domain Image-level detection performance on Impostor. For each method, ACC and F1 are reported in the upper and lower rows, respectively. From top to bottom, the baselines are grouped into generic classification backbones, conventional IMDL methods, and AIGC-oriented forensic detectors. **Bold** and underline denote the best and second-best results, respectively.

Method	Metrics	Add				Delete			Inpaint			Average
		Qwen-Image-Edit	FLUX-Kontext	Step-Edit	Nano-Banana	Qwen-Image-Edit	Omni Eraser	FLUX-Kontext	FLUX-Fill	Power-Paint	Qwen-Image-Edit	
ResNet-50	ACC	88.02	88.91	87.61	81.01	82.88	74.65	82.67	86.45	67.63	82.95	82.28
	F1	89.19	89.79	88.65	82.75	83.99	74.32	83.37	87.10	62.57	83.65	82.54
SwinV2-Tiny	ACC	96.42	96.40	96.15	97.37	95.92	95.14	96.08	96.47	83.93	96.63	95.05
	F1	96.46	96.47	96.23	97.39	96.01	95.15	96.12	96.53	81.65	96.68	94.87
MVSS-Net	ACC	89.62	88.73	86.77	82.61	85.82	79.34	85.84	86.94	78.42	86.75	85.08
	F1	88.44	89.87	88.18	84.24	87.26	80.20	86.97	87.99	78.65	88.09	85.99
ObjectFormer	ACC	82.36	81.80	81.32	76.71	81.23	79.21	81.64	83.19	81.26	82.09	81.08
	F1	85.00	84.60	84.26	81.10	84.19	82.63	84.47	85.60	83.46	84.81	84.01
TruFor	ACC	71.11	72.31	76.23	72.81	72.10	70.14	75.24	78.17	70.68	76.83	73.56
	F1	73.10	74.10	78.18	74.15	74.47	72.29	78.47	79.79	71.13	78.59	75.43
NFA-ViT	ACC	99.00	98.44	98.31	97.80	98.21	95.75	97.86	**98.52**	95.44	98.64	97.80
	F1	99.00	98.46	98.33	97.83	98.24	95.65	97.88	**98.54**	95.35	98.63	97.79
FatFormer	ACC	80.72	82.10	82.18	64.72	78.27	76.47	78.95	84.71	71.60	81.52	78.12
	F1	82.59	83.73	83.80	69.12	80.83	78.25	80.79	85.93	74.50	83.25	80.28
SPAI	ACC	72.41	72.86	71.98	71.64	72.27	73.15	72.79	72.93	71.38	72.52	72.39
	F1	75.46	75.82	75.01	74.59	75.34	76.10	75.76	75.91	74.32	75.58	75.39
AIDE	ACC	**99.25**	96.10	**99.48**	98.66	98.62	**99.10**	95.65	96.42	71.22	**98.68**	95.32
	F1	**99.25**	96.24	**99.48**	98.67	98.63	**99.10**	95.82	96.54	60.98	**98.69**	94.34
PANet	ACC	97.84	**99.69**	98.84	**98.84**	**98.84**	98.49	**98.76**	98.50	**97.32**	98.32	**98.54**
	F1	97.88	**99.70**	98.83	**98.85**	**98.85**	98.50	**98.77**	98.51	**97.31**	98.34	**98.55**

in Table 10, the performance gaps among different methods are also smaller than those observed in the cross-domain setting.

Nevertheless, several methods show clear performance drops on the PowerPaint (Zhuang et al., 2024) inpainting subset. For example, ResNet, Swin Transformer, and the AIGC-oriented detector AIDE (Yan et al., 2024) all perform noticeably worse on this subset. One possible reason is that PowerPaint produces visually coherent localized edits that introduce weaker global manipulation cues. In comparison, methods that jointly model localization and classification are more robust in this setting, as the localization objective encourages the network to focus on manipulated regions rather than relying only on global image-level signals. Consistent with this observation, PANet maintains high and stable performance across all subsets, including the particularly challenging PowerPaint dataset. Meanwhile, PANet also achieves the best overall average performance.

C IMPOSTOR DETAILS AND ANALYSIS

C.1 DETAILED STATISTICS AND DISTRIBUTION ANALYSIS

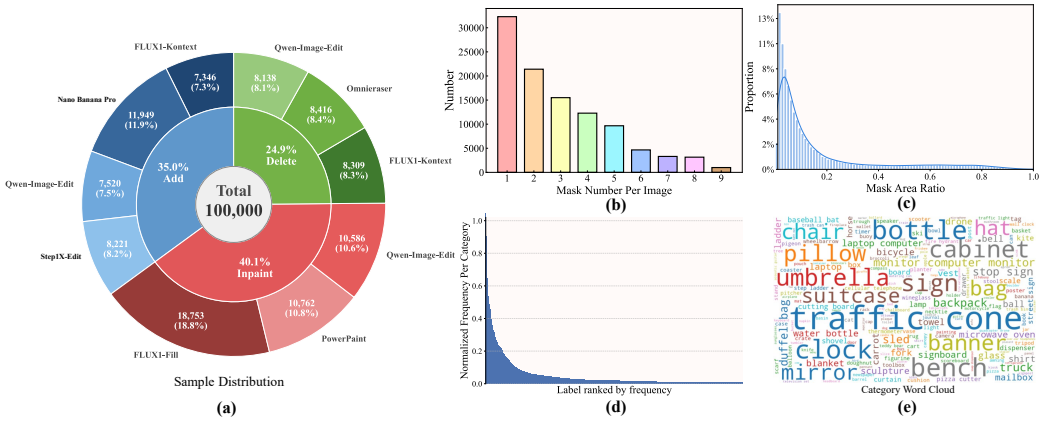

Figure 9: Dataset-level statistics for Impostor. (a) Sample distribution across manipulation types and generators; (b) number of manipulated regions per image; (c) manipulated-area ratio per image; (d) long-tail distribution of manipulated object categories ranked by normalized frequency; and (e) word cloud of manipulated object categories.

To further characterize Impostor, we analyze its dataset statistics along the three design objectives discussed in the main paper: higher visual realism, greater manipulation diversity, and broader generator coverage. **For visual realism**, the qualitative examples in Figure 15 illustrate that the manipulated content is visually coherent with the surrounding scene, exhibiting natural object boundaries, lighting, and texture transitions. This perceptual realism is supported by the DisAgent-based construction process. **For manipulation diversity**, Figure 9(a) shows that Impostor covers three representative manipulation types: object addition, removal, and inpainting. Figure 9(b) further shows both single-region and multi-region cases, with up to nine manipulated regions per image, covering more complex tampering scenarios. In addition, Figures 9(d) and 9(e) show that Impostor covers approximately 2,800 manipulated object categories, which exhibit a long-tail distribution. These statistics indicate that Impostor covers a broad range of manipulation scenarios in terms of editing type, region count, and semantic category. **For generator coverage**, Figure 9(a) shows that Impostor is constructed using seven advanced AIGC editing models. These models introduce diverse generation traces, helping reduce reliance on any single generator or artifact pattern. Overall, Impostor provides a more diverse, visually realistic, and challenging benchmark for AIGC-based manipulation localization, with higher visual realism, greater manipulation diversity, and broader generator coverage.

C.2 LVLM-BASED VISUAL REALISM ANALYSIS

Modern LVLMs have demonstrated strong capabilities in scene understanding, spatial reasoning, object-context modeling, and visual consistency assessment, making them a useful complement to human verification for evaluating the visual realism of manipulated images. To further verify that the low LVLM detection rates in Table 1 stem from the higher visual realism of Impostor rather than dataset-composition effects, we provide a fine-grained analysis of manipulated area, manipulation type, and manipulated-region count. All evaluations follow Prompt C.1, which instructs the models to identify AI-edited images solely based on observable visual evidence, including structural consistency, local texture details, boundary blending, and lighting/shadow coherence. Under this unified protocol, a lower detection rate indicates that the manipulated images expose fewer visible editing traces, providing additional evidence for stronger visual realism.

We first analyze the effect of manipulated area ratio, one of the most direct factors affecting visual detectability. Smaller manipulated regions can hide editing traces more easily, potentially leading to lower detection rates. We therefore select GPT-5.4(OpenAI, 2026) and Gemini 3 Pro(Google DeepMind, 2025b) as representative LVLM evaluators and perform area-stratified comparisons across datasets. As shown in Table 11, Impostor achieves the lowest detection rate in every manipulated-

Prompt C.1: LVLM-based Visual Realism Assessment

You are an expert in AI-generated content analysis and image forensics. Given an input image, determine whether it is an original **real camera-captured** image or an **AI-edited** image produced by tools such as Stable Diffusion, Midjourney, Photoshop, or similar generative editing systems.

Your decision must be based only on **observable visual evidence** in the image. Do not make the decision solely based on image content, image quality, artistic style, scene category, or subjective impression.

Please analyze the image from both **global consistency** and **local forensic details**. Specifically, consider the following aspects:

- **Global and Structural Consistency**: Check whether the overall scene layout, object arrangement, perspective, object scale, and spatial relationships are physically plausible and mutually consistent.
- **Local Texture and Detail**: Examine whether local regions contain abnormal textures, over-smoothed surfaces, distorted fine details, or inconsistent material appearance.
- **Boundary and Blending Artifacts**: Inspect object boundaries and transition areas for blurry contours, unnatural edge blending, or halo effects.
- **Lighting and Shadow Consistency**: Verify whether illumination direction, shadow shape, shadow intensity, and reflections are consistent across objects and regions.

Decision rule: If there is reliable visual evidence of editing traces or generation artifacts, label the image as **AI-edited**. If the image appears visually natural and there is no reliable observable evidence of AI editing, label it as **real camera-captured**.

Output requirement: Return only a valid JSON object.

Output format:

```
{
    "label": "<real camera-captured or AI-edited>",
    "reason" : "<one concise sentence summarizing the key visual evidence supporting the decision>"
}
```

Table 11: LVLM-based manipulated-image detection analysis stratified by manipulation area. Each area bin reports the detection rate (Det.) and the corresponding sample proportion (Prop.) of manipulated images in that bin. A lower Det. indicates harder-to-detect manipulations, suggesting fewer visible editing traces and stronger visual realism. Bold and underlined values denote the lowest and second-lowest Det. in each area bin, respectively.

Area Ratio	Metric	GPT-5.4					Gemini 3 Pro				
		CocoGlide	AutoSplice	SID-Set	BR-Gen	Impostor	CocoGlide	AutoSplice	SID-Set	BR-Gen	Impostor
0–4%	Det.	33.7%	37.7%	29.8%	<u>28.5%</u>	**15.6%**	<u>23.2%</u>	34.4%	28.9%	32.5%	**14.6%**
	Prop.	12.7%	2.2%	26.4%	6.3%	35.2%	12.7%	2.2%	26.4%	6.3%	35.2%
4–10%	Det.	50.1%	38.5%	33.5%	<u>32.3%</u>	**17.2%**	<u>33.9%</u>	38.7%	45.3%	43.7%	**19.8%**
	Prop.	24.8%	11.9%	22.2%	12.7%	28.7%	24.8%	11.9%	22.2%	12.7%	28.7%
10–20%	Det.	44.1%	52.0%	<u>37.3%</u>	37.9%	**22.1%**	<u>38.3%</u>	47.5%	57.4%	52.9%	**24.4%**
	Prop.	18.4%	12.8%	22.3%	14.3%	20.7%	18.4%	12.8%	22.3%	14.3%	20.7%
>20%	Det.	67.8%	69.0%	<u>48.9%</u>	52.3%	**24.4%**	76.1%	74.8%	<u>68.2%</u>	75.4%	**26.3%**
	Prop.	44.1%	73.1%	29.1%	66.7%	15.4%	44.1%	73.1%	29.1%	66.7%	15.4%
Overall	Det.	54.7%	62.5%	<u>37.9%</u>	46.2%	**18.8%**	52.0%	66.1%	<u>50.3%</u>	65.5%	**19.9%**

area bin for both LVLMs. Notably, even when the manipulated area exceeds 20%, where visual abnormalities are generally easier to observe, Impostor remains substantially harder to detect than the other datasets. These results show that the low LVLM detection rate of Impostor cannot be explained by smaller manipulated areas alone, but instead supports that Impostor exhibits higher visual realism with fewer visible editing traces.

We further analyze the effects of manipulation type and manipulated-region count within Impostor. Table 12 shows that Add is easier to detect than Delete and Inpaint, likely because inserted objects are more prone to boundary, lighting, or layout inconsistencies. Table 13 shows a similar trend for region count: samples with more manipulated regions are easier to detect because more edited regions increase the chance of exposing visible artifacts. These results indicate that the composition of Impostor does not favor low detection rates. In particular, Impostor contains a substantial proportion of Add samples, and more than half of its samples involve multiple manipulated regions.

Table 12: LVLM-based manipulated-image detection analysis of Impostor stratified by manipulation type. Columns correspond to different manipulation types, with the sample proportion shown below each type.

Model / Manip. Type	Inpaint 30.0%	Add 40.0%	Delete 30%
Qwen3VL	11.7%	17.1%	10.4%
Gemini3Pro	12.6%	30.2%	13.3%
Fakeshield	7.7%	12.6%	9.6%
GPT-5.4	16.1%	26.1%	11.6%

Table 13: LVLM-based manipulated-image detection analysis on Impostor stratified by manipulated-region count. Columns correspond to different numbers of manipulated regions per image, with the sample proportion shown below each region count.

Model / Manip. Region	1 42.9%	2 22.4%	3 13.1%	4 11.5%	$\geq$5 10.1%
Qwen3VL	7.6%	13.9%	17.3%	19.6%	25.5%
Gemini3Pro	16.4%	14.5%	22.6%	27.5%	34.1%
Fakeshield	7.1%	10.5%	11.3%	13.2%	18.2%
GPT-5.4	11.8%	19.2%	22.3%	26.1%	34.3%

Taken together, these analyses indicate that the low LVLM detection rate of Impostor is not primarily driven by manipulated area, manipulation type, or region-count composition, but instead reflects fewer visible editing traces and higher visual realism.

C.3 Category Definition and Quantification Protocol

To enable a unified comparison of manipulated semantic diversity across different benchmarks, we adopt an approximate protocol to define and quantify manipulation categories. A manipulation category refers to the fundamental semantic class of the manipulated object. Specifically, for object addition, the category corresponds to the added object; for object removal, it corresponds to the removed object; for object inpainting, it corresponds to the edited target object after manipulation. We introduce this protocol because limited manipulated-category diversity may encourage models to rely on recurrent semantic object priors rather than learning transferable manipulation cues. We therefore hypothesize that richer manipulated semantic diversity can promote stronger cross-dataset generalization. Consistent with this hypothesis, the results in Table 7 support our observation that richer manipulated semantic diversity is associated with stronger cross-dataset generalization.

For datasets with category metadata, we directly obtain the region-category correspondence from the provided annotation files. For datasets without explicit category metadata, we perform approximate category inference using a unified visual recognition pipeline. Let M denote the manipulation mask of an image, and let $n = \mathcal{N}_{\mathrm{cc}}(M)$ denote the number of connected components in M, which provides an approximate estimate of the number of manipulated instances. We then apply RAM to the manipulated regions to obtain a candidate label set:

$$\mathcal{L} = \{l_1, l_2, \ldots, l_K\}, \tag{16}$$

where K is the number of candidate categories proposed by RAM. Each candidate label l_k is further verified by Grounded DINO, producing a grounding confidence score s_k. We then rank all verified candidates by confidence and retain the top-n labels:

$$\hat{\mathcal{L}} = \text{Top-}n\Big(\{(l_k, s_k)\}_{k=1}^{K}\Big). \tag{17}$$

The retained labels are used as the manipulated category assignments for the image. Since this procedure relies on visual recognition and connected-component estimation rather than explicit semantic annotations, the resulting category statistics should be regarded as approximate reference measurements for cross-benchmark comparison.

To improve statistical consistency, we further normalize all extracted labels using an LLM. Attributes such as colors, adjectives, and other modifiers are removed, while the basic category names are retained. Let $\phi(\cdot)$ denote this normalization function. For an image i, let $\hat{\mathcal{L}}_i$ denote its manipulated category set obtained either from annotations or from the inference procedure above. The corresponding normalized category set is defined as

$$\mathcal{C}_i = \{\phi(l) \mid l \in \hat{\mathcal{L}}_i\}. \tag{18}$$

For a dataset containing N images, category coverage is defined as the total number of unique normalized categories: $\text{Coverage} = \left|\bigcup_{i=1}^{N} \mathcal{C}_i\right|$.

To suppress potential noise introduced by parsing or recognition errors, singleton categories that appear only once in the entire dataset are excluded from the final reported statistics.

D Details of the Training Losses

D.1 Forensic-Semantic Consistency Loss

The Forensic-Semantic Consistency (FSC) loss is designed to explicitly supervise the local consistency between semantic representations and manipulation-sensitive forensic cues. In authentic images, high-level semantic structures and low-level forensic traces are jointly constrained by the same natural image formation process, resulting in a relatively stable correspondence between their representations. In contrast, recent AIGC editing models can produce semantically plausible local edits (Labs et al., 2025; Wu et al., 2025), while the editing process may disrupt low-level forensic traces (Chai et al., 2020; Zhong et al., 2024), thereby weakening such correspondence. Our diagnostic analysis in Sec. A.1 further provides empirical support for this hypothesis, showing that authentic regions generally exhibit stronger forensic-semantic consistency than manipulated regions. Importantly, FSC does not assume that every manipulated region necessarily exhibits low consistency. Instead, it encourages authentic regions to produce stronger FSC responses while enlarging the relative consistency gap between authentic and manipulated regions.

FSC Module Given the semantic feature $F_s \in \mathbb{R}^{B\times C_s\times H\times W}$ and forensic feature $F_f \in \mathbb{R}^{B\times C_f\times H\times W}$, we project them into a shared D-dimensional embedding space using two lightweight 1×1 convolutional projection heads, followed by channel-wise L_2 normalization:

$$F_{\delta,s} = \mathrm{L2Norm}\big(\phi_s(F_s)\big), \qquad F_{\delta,f} = \mathrm{L2Norm}\big(\phi_f(F_f)\big), \tag{19}$$

where $\phi_s(\cdot)$ and $\phi_f(\cdot)$ denote the semantic and forensic projection heads, respectively, and $F_{\delta,s}, F_{\delta,f} \in \mathbb{R}^{B\times D\times H\times W}$. The local forensic-semantic consistency score is then computed as the cosine similarity between the projected features:

$$C(x,y) = \mathrm{CosSim}\left(F_{\delta,s}(:,x,y), F_{\delta,f}(:,x,y)\right), \qquad C \in \mathbb{R}^{B\times 1\times H\times W}, \tag{20}$$

where a larger $C(x,y)$ indicates stronger local forensic-semantic consistency.

FSC Loss. We design two complementary constraints. The first term $\mathcal{L}_{\mathrm{real}}$ encourages authentic regions to produce high forensic-semantic consistency (FSC) responses, while the second term $\mathcal{L}_{\mathrm{sep}}$ enlarges the relative consistency gap between authentic and manipulated regions.

Let Ω_{real} and Ω_{fake} denote the authentic and manipulated regions defined by the ground-truth mask, respectively. To encourage high FSC in authentic regions, we impose positive consistency supervision:

$$\mathcal{L}_{\mathrm{real}} = \frac{1}{|\Omega_{\mathrm{real}}|} \sum_{i\in\Omega_{\mathrm{real}}} \mathrm{BCEWithLogits}(\tau C_i, 1), \tag{21}$$

where τ is a learnable scaling factor. This term encourages authentic regions to produce strong FSC responses. Since manipulated regions may still preserve partial forensic-semantic consistency, we do not explicitly force every manipulated location to produce a low FSC score. Instead, we constrain the relative consistency gap between the authentic and manipulated regions. Specifically, their average FSC scores are computed as

$$\bar{C}_{\mathrm{real}} = \frac{1}{|\Omega_{\mathrm{real}}|} \sum_{i\in\Omega_{\mathrm{real}}} C_i, \qquad \bar{C}_{\mathrm{fake}} = \frac{1}{|\Omega_{\mathrm{fake}}|} \sum_{i\in\Omega_{\mathrm{fake}}} C_i. \tag{22}$$

We then introduce a margin-based consistency separation loss:

$$\mathcal{L}_{\mathrm{sep}} = \max\left(0, \delta - \left(\bar{C}_{\mathrm{real}} - \bar{C}_{\mathrm{fake}}\right)\right), \tag{23}$$

where δ denotes the margin. The overall FSC loss is defined as

$$\mathcal{L}_{\mathrm{fsc}} = \mathcal{L}_{\mathrm{real}} + \mathcal{L}_{\mathrm{sep}}. \tag{24}$$

D.2 TraceNCE Loss

To improve the separability between authentic and manipulated representations in the decoder space, we introduce a lightweight contrastive loss, termed TraceNCE. Given the decoder feature map, we

first downsample the annotation mask to the feature resolution and sample feature vectors from authentic and manipulated regions. These sampled features are then projected into an embedding space for contrastive learning. TraceNCE encourages features from the same region type to be closer while separating features from different region types.

Formally, let $\mathcal{P} = \{p_i\}_{i=1}^{N}$ denote the sampled projected features, and let $y_i \in \{0, 1\}$ denote the corresponding authentic/manipulated label. For each valid anchor i, TraceNCE is defined as

$$\mathcal{L}_{\text{TraceNCE}} = -\frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \log \frac{\sum\limits_{j \neq i,\ y_j = y_i} \exp\left(\text{sim}(p_i, p_j)/t\right)}{\sum\limits_{k \neq i} \exp\left(\text{sim}(p_i, p_k)/t\right)}, \tag{25}$$

where $\text{sim}(\cdot, \cdot)$ denotes cosine similarity, t is the temperature parameter, and $\mathcal{I}$ denotes the set of anchors with at least one positive sample. TraceNCE complements FSC by enhancing region-level discrimination between authentic and manipulated representations in the decoder space.

D.3 Application to Image-Level Detection.

For image-level training on ProGAN, we construct each mini-batch with equal numbers of authentic and fully generated images and assign each image label to all spatial locations of its feature maps. FSC is computed over the mini-batch rather than separately for each image. Specifically, Ω_{real} and Ω_{fake} contain the spatial locations from authentic and generated images within the same mini-batch, respectively, with i indexing both images and spatial locations. Each local consistency score is still computed between the semantic and forensic features at the corresponding location within the same image. The class means are normalized by their respective numbers of locations, while $\mathcal{L}_{\text{real}}$ is averaged over authentic locations. Both sets are therefore nonempty, and $\mathcal{L}_{\text{sep}}$ compares their mean consistency without requiring every generated location to have low consistency. For TraceNCE, $\mathcal{P}$ pools decoder features sampled from both classes within the same mini-batch, with each feature inheriting its image label. Other sampled features with the same label serve as positives, and those with the opposite label serve as negatives. Self-pairs are excluded, and the loss is averaged over valid anchors as defined above. Thus, neither loss requires authentic and generated regions to coexist within an individual training image.

E Impostor Creation Pipeline

E.1 Pipeline Overview

As illustrated in Figure 2, Impostor is constructed through a three-stage pipeline consisting of raw input preparation, an autonomous DisAgent loop, and human verification. Within the DisAgent loop, the perception stage first determines the manipulation type, while the planning stage specifies where and how the image should be edited. The execution stage then invokes suitable editing models to generate candidate manipulated images, followed by the validation stage, which evaluates each candidate according to four quality criteria. The resulting feedback is passed to the reflection stage to summarize failure patterns, update the status memory, and guide subsequent perception and planning, thereby forming a closed-loop generation process. After automatic generation and filtering, the retained candidates undergo human verification for final curation.

To evaluate the contribution of the closed-loop design in DisAgent, Table 14 compares four construction settings before the final human-verification stage, while keeping perception, planning, and execution unchanged. The *Sequential Pipeline* removes both validation and reflection, resulting in a sequential construction process without sample-level quality filtering or feedback to subsequent planning. In the *w/o Validation* setting, sample-level quality validation is disabled, while reflection is retained and updates the historical memory using only statistics collected from the generated dataset, such as manipulation type, manipulated area, object count, and category distribution. In the *w/o Reflection* setting, validation is retained to filter low-quality or inconsistent samples, but validation outcomes and dataset statistics are not summarized into historical memory or propagated to subsequent perception and planning. The full DisAgent incorporates both validation and reflection, allowing quality-control results and dataset-level statistics to jointly guide subsequent generation. The results show a clear progressive improvement from the sequential pipeline to the complete

Table 14: Impact of the Agent-driven construction pipeline on image-level detection and pixel-level localization. Lower values indicate more challenging manipulated samples.

Setting	Image-level Detection ACC				Pixel-level Localization IoU			
	Qwen3VL	FakeShield	Gemini3Pro	GPT-5.4	FakeShield	MVSS-Net	NFA-ViT	Sparse-ViT
Sequential Pipeline	33.81	25.92	40.25	34.71	20.24	8.04	20.47	9.48
w/o Validation	30.37	23.48	35.21	29.45	18.88	7.98	19.83	8.35
w/o Reflection	20.61	14.23	27.04	24.93	16.65	7.78	19.05	7.30
w/ Agent	17.91	12.16	24.56	22.84	15.98	7.51	18.71	6.91

closed-loop design, with validation providing the main quality-control effect and reflection offering complementary gains through iterative feedback and distribution regulation. Figure 12 further shows that reflection produces a more balanced manipulated-category distribution.

E.2 Perception: Editability Assessment

The perception stage acts as the task-routing step in DisAgent, selecting a suitable manipulation type for each image based on its scene content and editing feasibility. Formally, given an input image I and the historical status memory $\mathcal{H}$, this task-routing process is formulated as:

$$t^* = \Phi_{\text{perc}}(I, \mathcal{H}), \qquad t^* \in \{\text{add}, \text{delete}, \text{inpaint}\}, \tag{26}$$

where $\Phi_{\text{perc}}(\cdot)$ denotes the perception module, and the three labels correspond to object addition, object removal, and object inpainting, respectively. The module analyzes scene semantics, object composition, and background context to determine which manipulation type is the most suitable for the image.

This design reduces unsuitable editing choices at the beginning of the pipeline and provides a reliable basis for subsequent planning, execution, and validation. In addition, the historical status memory $\mathcal{H}$ guides this task routing by tracking the distribution of manipulation types, thereby reducing bias toward any single manipulation type.

E.3 Planning: Mask-Prompt Co-Planning

High-quality localized editing requires close alignment between editing instructions and target regions. Existing benchmark-construction pipelines (Cai et al., 2026; Huang et al., 2025) typically generate editing instructions from image-level captions or object categories, without explicitly incorporating fine-grained information about the editing mask, such as its geometry, spatial layout, and surrounding context. This may result in mismatches between editing instructions and target regions, especially in complex localized editing scenarios. To address this limitation, our planning stage performs mask-prompt co-planning through two coupled steps: it first identifies suitable editing regions and then generates editing instructions conditioned on the selected regions. By grounding instruction generation on the selected regions, this design reduces mask-instruction mismatch. The planning process is driven by advanced LVLMs, including ChatGPT-5.1 (OpenAI, 2025) and Qwen3-VL-235B-A22B (Bai et al., 2025a), which provide multimodal reasoning for editing-region determination and fine-grained instruction generation.

For *delete* and *inpaint*, DisAgent first selects target editing masks from candidate regions and then generates editing instructions conditioned on the selected regions and their local context. For *add*, no pre-existing target mask is required because the goal is to insert new content into the image. Instead, DisAgent determines a suitable insertion area and generates a region-compatible instruction. After the addition edit, the corresponding tampering mask is obtained by localizing and segmenting the inserted object with Grounded SAM 2 (Ren et al., 2024). In both branches, the planning process is further guided by the historical status memory maintained by the reflection stage, which helps improve instruction plausibility and maintain dataset balance throughout the generation process.

Delete / Inpaint Branch. For *delete* and *inpaint* operations, the planning stage takes the input image I, a set of candidate masks $\mathcal{M} = \{m_i\}_{i=1}^{N}$, the historical status memory $\mathcal{H}$, and the manipulation

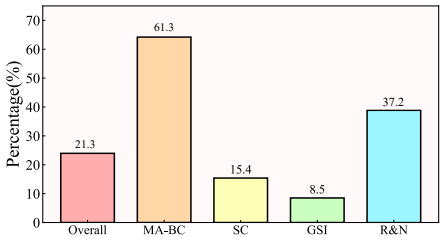

Figure 10: Overall filtering ratio and failure-cause distribution.

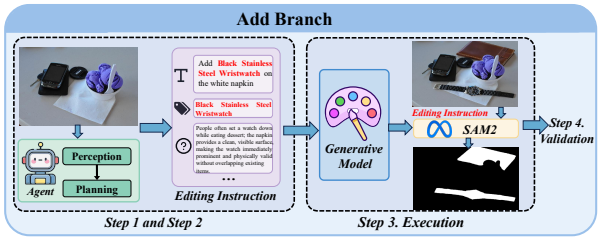

Figure 11: Add branch pipeline with post-hoc tampering mask generation via Grounded SAM 2 (Ren et al., 2024).

Table 15: Summary of validation criteria in DisAgent.

Criterion	Purpose	Measurement	Pass Rule
MA-BC	Ensures that the target region changes sufficiently while the background remains consistent with the source image.	DINOv2/ResNet feature similarity inside and outside the mask, denoted as $\mathrm{sim}_{\mathrm{in}}$ and $\mathrm{sim}_{\mathrm{out}}$.	Retain if $\mathrm{sim}_{\mathrm{in}} \leq 0.60$ and $\mathrm{sim}_{\mathrm{out}} \geq 0.80$.
SC	Ensures that the edited content is semantically consistent with the intended target category.	CLIP cosine similarity sim_{sem} between the manipulated region and the target category.	Retain if $sim_{\mathrm{sem}} \geq 0.23$.
GSI	Ensures that the manipulated region is structurally plausible and free of obvious local discontinuities.	MANIQA quality score q_{gsi} on the target-region crop.	Retain if $q \geq 0.45$.
R&N	Ensures that the manipulated image appears natural and free of obvious editing artifacts.	LVLM-based visual realism judgment using Prompt D.1.	Retain if the sample is judged visually realistic.

type t^* determined by the perception stage as input. It selects suitable editing masks and generates the corresponding editing instruction:

$$(\mathcal{M}^*, p^*) = \Phi_{\mathrm{plan}}(I, \mathcal{M}, \mathcal{H}, t^*), \qquad t^* \in \{\mathrm{delete}, \mathrm{inpaint}\}, \tag{27}$$

where $\mathcal{M}^* \subseteq \mathcal{M}$ denotes the selected set of editing masks, and p^* denotes the editing instruction.

Add Branch. For *add*, the goal is to insert new content into the image rather than modify an existing object. Therefore, no pre-existing object mask is required. The planning stage generates an addition instruction based on the image content and historical status memory:

$$p^* = \Phi_{\mathrm{add}}(I, \mathcal{H}, t^*), \qquad t^* = \mathrm{add}, \tag{28}$$

where p^* specifies the category, location, and visual attributes of the object to be added, with consideration of the surrounding objects, background, and scene layout.

E.4 Execution: Multi-Generator Manipulation Construction

Given the input image I, the editing instruction p^*, the manipulation type t^*, and the optional editing masks $\mathcal{M}^*$, the execution stage selects a suitable editing model from the corresponding generator pool to generate the manipulated image:

$$\hat{I}, \mathcal{G} = \Phi_{\mathrm{generation}}(I, p^*, t^*, \mathcal{M}^*), \tag{29}$$

where $\hat{I}$ denotes the manipulated image and $\mathcal{G}$ denotes the generation metadata. The editing masks $\mathcal{M}^*$ are provided for *delete* and *inpaint*, while they are omitted for *add*, where new content is introduced without a pre-existing target mask.

For *add*, the corresponding tampering masks are obtained after generation. Specifically, Grounded SAM 2 (Ren et al., 2024) localizes and segments the inserted objects according to the object-category information specified in p^*:

$$\mathcal{M}^* = \Phi_{\mathrm{GroundedSAM2}}\big(\hat{I}, c(p^*)\big), \tag{30}$$

where $c(p^*)$ denotes the object-category information extracted from the editing instruction. The execution stage also records the editing instruction, manipulation type, generator identity, and generation parameters as metadata for subsequent validation and reflection.

E.5 VALIDATION: AUTOMATIC QUALITY FILTERING

To ensure the quality of samples generated by the automated DisAgent pipeline, we perform automatic filtering using four criteria: Semantic Consistency (SC), Mask Alignment and Background Consistency (MA-BC), Geometric and Structural Integrity (GSI), and overall Realism and Naturalness (R&N). Table 15 summarizes the measurements and pass rules for these validation criteria. Figure 10 reports both the overall filtering ratio and the criterion-wise trigger distribution of different quality issues. The overall filtering ratio denotes the proportion of rejected samples among all generated samples. For rejected samples, we further compute the proportion triggered by each criterion. Since these criteria are not mutually exclusive, a single sample may trigger multiple filtering conditions, and thus the sum of all proportions may exceed 100%. As shown in Figure 10, MA-BC exhibits the highest trigger rate, suggesting that generation failures mainly stem from unstable region control. This issue is particularly challenging in multi-region editing, which requires simultaneous control over multiple spatially separated regions and therefore increases the likelihood of local mismatches and inter-region interference. Under our validation rule, a sample is rejected if any edited region violates the region-consistency constraint. In contrast, the low trigger rate of GSI demonstrates the effectiveness of our choice of generation models, which generally produce structurally coherent content with fewer severe geometric distortions or structural artifacts. Representative rejected samples under different validation criteria are shown in Figure 13.

Mask Alignment / Background Consistency (MA-BC). This criterion is designed to ensure that manipulations are confined to the intended editing region while preserving the surrounding background. Specifically, the background should remain highly consistent with the source image, whereas the manipulated region should exhibit sufficient changes to confirm that the intended edit has taken place.

To measure this property, we independently extract intermediate features from the source and manipulated images using DINOv2-Base (Oquab et al., 2023) and ResNet-50 (He et al., 2016). The MA-BC procedure is applied separately to the two backbones. For each backbone, let the feature maps of the source and manipulated images be denoted by $f_s, f_m \in \mathbb{R}^{D\times H_f\times W_f}$, where D is the feature dimension and $H_f \times W_f$ is the spatial resolution. We compute a spatial similarity map using the cosine similarity between corresponding feature vectors:

$$\mathrm{heat}(x, y) = \mathrm{CosSim}\left(f_s(:, x, y), f_m(:, x, y)\right), \tag{31}$$

where $\mathrm{heat} \in \mathbb{R}^{H_f\times W_f}$ measures the similarity between the source and manipulated images at each spatial location. Let $M_t \in \{0, 1\}^{H_f\times W_f}$ denote the resized manipulation mask, where 1 indicates manipulated regions and 0 indicates pristine regions. For each backbone, we compute the average similarity outside and inside the manipulated region as:

$$\mathrm{sim}_{\mathrm{out}} = \frac{\sum\left(\mathrm{heat}\cdot(1-M_t)\right)}{H_f W_f - \sum M_t + \epsilon}, \qquad \mathrm{sim}_{\mathrm{in}} = \frac{\sum\left(\mathrm{heat}\cdot M_t\right)}{\sum M_t + \epsilon}. \tag{32}$$

Here, $\mathrm{sim}_{\mathrm{out}}$ measures the source–manipulated similarity outside the editing region, while $\mathrm{sim}_{\mathrm{in}}$ measures the similarity inside the editing region. A valid manipulation should preserve high similarity outside the target region while introducing sufficient changes within it.

We apply the above criterion independently to DINOv2-Base and ResNet-50. For each backbone, a sample fails MA-BC if $\mathrm{sim}_{\mathrm{out}} < 0.80$ or $\mathrm{sim}_{\mathrm{in}} > 0.60$. The sample is rejected if either backbone reports a failure. This rule removes cases with unintended background changes or insufficient modification of the target region.

Semantic Consistency (SC). In Impostor, the metadata records the manipulation mask and corresponding editing instruction for each manipulated region. Based on this information, we evaluate whether the manipulated content is semantically aligned with the intended target category using CLIP (Radford et al., 2021). For each manipulated region, we compute the cosine similarity between the visual feature f_v extracted from the manipulated region and the text feature f_t of the target object category:

$$\mathrm{sim}_{\mathrm{semantic}} = \mathrm{CosSim}(f_v, f_t). \tag{33}$$

A manipulated region passes the SC criterion if $\mathrm{sim}_{\mathrm{semantic}} \geq 0.23$. For samples containing multiple manipulated regions, the sample is rejected if any region fails this criterion. This filtering step removes edits whose generated content is inconsistent with the intended manipulation target.

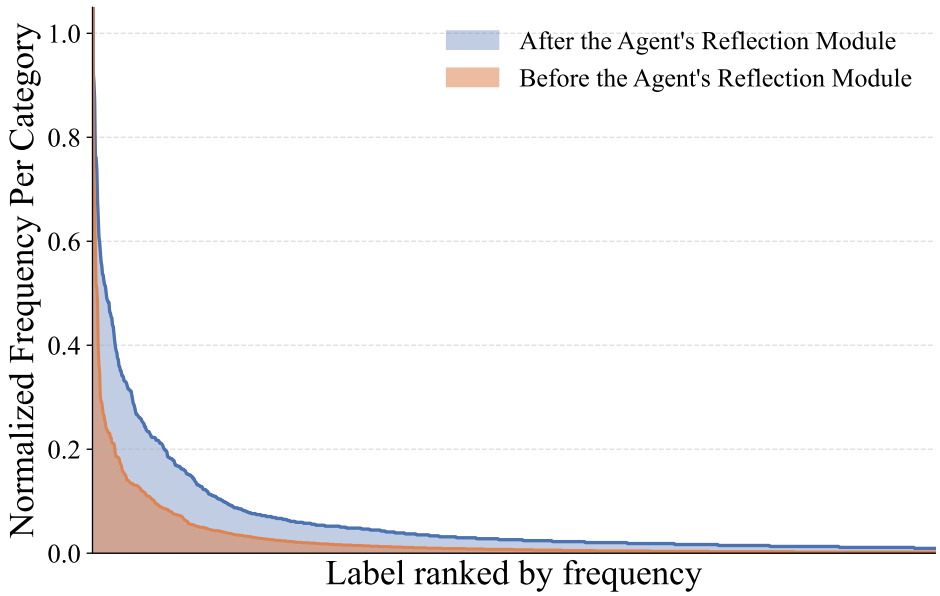

Figure 12: Comparison of manipulated object category distributions before and after applying the Agent's Reflection Module. After reflection, the distribution becomes more balanced, alleviating the dominance of high-frequency categories.

To automatically filter out generated samples with structurally implausible manipulated regions, such as object fragmentation, local discontinuities, or micro-artifacts, we adopt the no-reference image quality assessment model MANIQA (Yang et al., 2022). MANIQA models perceptual image quality through multi-dimension attention, making it well suited for detecting structural distortions and unnatural local patterns introduced by generative editing. For each manipulated region, we compute:

$$q = \mathrm{MANIQA}(\hat{I}_{\mathrm{patch}}). \tag{34}$$

A manipulated region passes the GSI criterion if $q \geq 0.45$. For samples containing multiple manipulated regions, the sample is rejected if any region fails this criterion.

Realism & Naturalness (R&N). To assess the realism and naturalness of manipulated images, we leverage Qwen2.5-VL 7B (Bai et al., 2025b) to evaluate lighting consistency, texture realism, scene coherence, and geometric plausibility. LVLMs are well-suited for detecting subtle inconsistencies in illumination, textures, and object geometry that conventional metrics may overlook.

E.6 Reflection: Failure Attribution and Iterative Update

The reflection module is driven by a separate LLM and performs three main tasks: summarizing statistics of successful samples, attributing failure causes of rejected samples, and generating feedback constraints for subsequent iterations. For successful samples, DisAgent records global status information, including the manipulated area, number of manipulated objects, and category distribution, in the historical status memory $\mathcal{H}$. For rejected samples, the validation results are provided to the LLM for failure attribution. Based on these statistics and failure analyses, the LLM generates feedback constraints and updates $\mathcal{H}$, which are then used to guide subsequent perception and planning. This iterative feedback mechanism helps reduce recurrent failure patterns and maintain a more balanced generation distribution over time. As shown in Figure 12, reflection produces a more balanced object-category distribution by reducing the dominance of high-frequency categories.

F Experimental Details: Compared Methods, Datasets, and Splits

This section provides supplementary details on the methods and datasets involved in the experiments, including the evaluated methods, the datasets used, and the corresponding data split protocols.

F.1 Compared Methods

We summarize the compared methods involved in our experiments from two aspects: image-level manipulation detection and pixel-level manipulation localization. The included baselines include both conventional forgery analysis methods and recent AIGC-oriented methods.

F.1.1 LVLM-based Visual Realism Evaluators

In Table 1, we utilize LVLMs as zero-shot evaluators to assess the visual realism of manipulated images from different datasets. Given a manipulated image, each LVLM is asked to determine whether the image has been edited. A lower detection rate therefore indicates higher visual realism and greater difficulty of the corresponding dataset. Below, we provide the detailed information of the LVLMs used in this evaluation.

- **Qwen3-VL-235B-A22B-Instruct** (2025) (Bai et al., 2025a) is a large-scale open-source LVLM with strong visual understanding and multimodal reasoning capabilities. We use the 235B-A22B Instruct version as the open-source general-purpose evaluator.
- **FakeShield-22B** (2025) (Xu et al., 2025) is a forgery-oriented LVLM designed for image forgery detection, localization, and explanation. We use the 22B model as the specialized forensic evaluator to examine how specialized LVLM performs on Impostor.
- **Gemini 3 Pro** (2025) (Google DeepMind, 2025b) is a proprietary LVLM with strong image understanding and reasoning ability. We use it as a closed-source commercial evaluator.
- **GPT-5.4** (2026) (OpenAI, 2026) is a proprietary frontier reasoning model with strong vision and reasoning capabilities. We use it as another closed-source commercial evaluator.

F.1.2 Image-level detection methods

- **CNNSpot** (CVPR'2020) (Wang et al., 2020) learns a standard image classifier with carefully designed pre- and post-processing augmentations, and shows that such a simple detector can generalize surprisingly well to unseen CNN/GAN generators.
- **FreDect** (ICML'2020) (Frank et al., 2020) exploits frequency-domain artifacts left by GAN synthesis and uses them as discriminative cues for fake-image classification.
- **Fusing** (ICIP'2022) (Ju et al., 2022) adopts a two-branch architecture that combines global image content with informative local patches, and further fuses them with attention. This design improves robustness to content variation and enhances generalization to diverse synthetic images.
- **Gram-Net** (CVPR'2020) (Liu et al., 2020) learns global texture statistics through Gram-matrix-style representations, motivated by the observation that synthetic images exhibit texture discrepancies from real ones. It is especially designed to capture transferable texture artifacts and improve robustness under common image degradations.
- **LNP** (ECCV'2022) (Liu et al., 2022a) models learnable noise patterns extracted from images and uses them for fake/real discrimination. By emphasizing low-level noise cues rather than image semantics, it achieves strong generalization across multiple generative models.
- **UnivFD** (CVPR'2023) (Ojha et al., 2023) builds a universal fake-image detector on top of pretrained CLIP features, showing that a simple linear classifier over strong frozen representations can generalize well across generators.
- **DIRE** (ICCV'2023) (Wang et al., 2023) introduces diffusion reconstruction error as a discriminative representation, based on the observation that diffusion-generated images can often be reconstructed by a diffusion model more faithfully than real images.
- **PatchCraft** (Arxiv'2024) (Zhong et al., 2024) enhances texture-sensitive regions through a Smash&Reconstruction preprocessing step and further identifies fake images using the contrast of inter-pixel correlations between rich- and poor-texture patches.
- **NPR** (CVPR'2024) (Tan et al., 2024) rethinks the artifacts introduced by up-sampling operations in CNN-based generators and proposes neighboring pixel relationships as a simple yet effective generalized artifact representation.

- **AIDE** (ICLR'2025) (Yan et al., 2024) combines CLIP-based semantic features with noise-aware cues from frequency-selected patches and SRM filtering, improving cross-generator generalization for AIGC image detection.

F.1.3 PIXEL-LEVEL LOCALIZATION METHODS.

- **MVSS-Net** (ICCV'2021) (Dong et al., 2022) learns manipulation cues through multi-view feature learning by jointly exploiting boundary artifacts and noise-view information, and is trained with multi-scale supervision over pixel-, edge-, and image-level targets.
- **CAT-NET** (IJCV'2022) (Kwon et al., 2022) localizes manipulated regions by tracing JPEG compression artifacts with a dual-stream design that processes RGB content and DCT-coefficient representations jointly.
- **ObjectFormer** (CVPR'2022) (Wang et al., 2022) detects and localizes manipulated regions by fusing RGB features with high-frequency forensic features as multimodal patch embeddings, and uses learnable object prototypes to model object-level consistency across regions for refining patch-level manipulation representations.
- **TruFor** (CVPR'2023) (Guillaro et al., 2023) performs image forgery detection and localization by leveraging a learned noise-sensitive fingerprint together with complementary forensic cues, and further predicts a confidence map for trustworthy analysis.
- **IML-VIT** (arXiv'2024) (Ma et al., 2023) is an end-to-end vision-transformer-based model for image manipulation localization, built to handle high-resolution inputs while modeling multiple discrepancies relevant to manipulated regions.
- **Mesorch** (AAAI'2025) (Zhu et al., 2025) introduces a mesoscopic perspective for image manipulation localization by orchestrating multi-scale representations with a hybrid architecture to jointly model microscopic traces and macroscopic object-level cues.
- **M2SFormer** (ICCV'2025) (Nam et al., 2025) is a transformer-based framework that unifies multi-spectral and multi-scale attention, and further incorporates edge-aware difficulty guidance to preserve subtle forgery details during localization.
- **NFA-ViT** (AAAI'2026) (Cai et al., 2026) introduces a noise-guided forgery amplification transformer with a dual-branch design, aiming to enhance weak manipulation cues and improve the localization of subtle localized AIGC forgeries.
- **RelayFormer** (ICLR'2026) (Huang et al., 2026) introduces a unified local–global attention framework with relay tokens, enabling efficient manipulation localization across varying image resolutions while preserving fine-grained forensic cues.
- **RITA** (CVPR'2026) (Zhu et al., 2026) reformulates manipulation localization as a conditional sequence prediction task, progressively localizing manipulated regions to explicitly model the sequential and hierarchical dependencies among multiple editing operations.
- **FLAME** (ICML'2026) (Wang et al., 2026) exploits local statistical energy anomalies induced by generative editing through a Local Anomaly Distribution (LAD) representation, together with a parameter-efficient SAM adapter for precise manipulation localization.

F.2 COMPARED DATASETS AND SPLITS

We summarize the traditional and AIGC-based IMDL datasets used in our experiments and clarify their roles under different evaluation settings.

F.2.1 TRADITIONAL IMAGE MANIPULATION LOCALIZATION DATASETS.

- **Columbia** (Ng et al., 2004) is an early image splicing benchmark released by Columbia University, containing authentic and spliced image samples for IMDL studies.
- **CASIA V1** (Dong et al., 2013) is a classical tampering benchmark focusing on splicing manipulations, constructed from natural images with corresponding authentic and tampered pairs.
- **CASIA V2** (Dong et al., 2013) extends CASIA V1 with a larger scale and more diverse tampering types, including both splicing and copy-move manipulations, and is widely used for image manipulation localization evaluation.

- **PSCC-Net** (Liu et al., 2022b) denotes the splicing dataset released with PSCC-Net, consisting of a content-aware copy-paste subset generated from MS COCO object masks and a random-mask copy-paste subset constructed from multiple real-image sources.
- **SP COCO** (Kwon et al., 2022) is a COCO-based JPEG splicing dataset built by pasting annotated objects into other images and resaving them under JPEG compression. It is used in CAT-Net to study splicing localization with compression-artifact cues in RGB and DCT domains.
- **IMD2020** (Novozamsky et al., 2020) is a real-life image manipulation dataset collected from the Internet, providing corresponding authentic images and manually annotated masks for localized tampering regions.

F.2.2 AIGC IMAGE MANIPULATION LOCALIZATION DATASETS.

- **CocoGlide** (Guillaro et al., 2023) is an AIGC-oriented localized manipulation dataset built on real images with edited regions generated by text-guided diffusion-based editing, and is commonly used to evaluate localized AIGC manipulation localization.
- **AutoSplice** (Jia et al., 2023) is a text-prompt manipulated image dataset constructed with DALL-E2-guided local or global editing on real-world image-caption pairs, and supports both forgery detection and localization tasks.
- **SID-Set** (Huang et al., 2025) is a large-scale social-media-oriented benchmark containing authentic, synthetic, and tampered images, with annotations for image-level detection, pixel-level localization, and textual explanation.
- **BR-Gen** (Cai et al., 2026) is a large-scale benchmark for localized AIGC manipulation analysis, designed to cover more realistic local forgeries beyond object-centric edits, including subtle manipulations over stuff and background regions.
- **OpenSDI** (Wang et al., 2025) is a large-scale open-world benchmark for diffusion-generated image detection, covering both image-level detection and pixel-level localization across multiple diffusion models.

G THE VISUALIZATION OF SAMPLES

G.1 VISUALIZATION OF REPRESENTATIVE REJECTED SAMPLES DURING AUTOMATIC QUALITY FILTERING

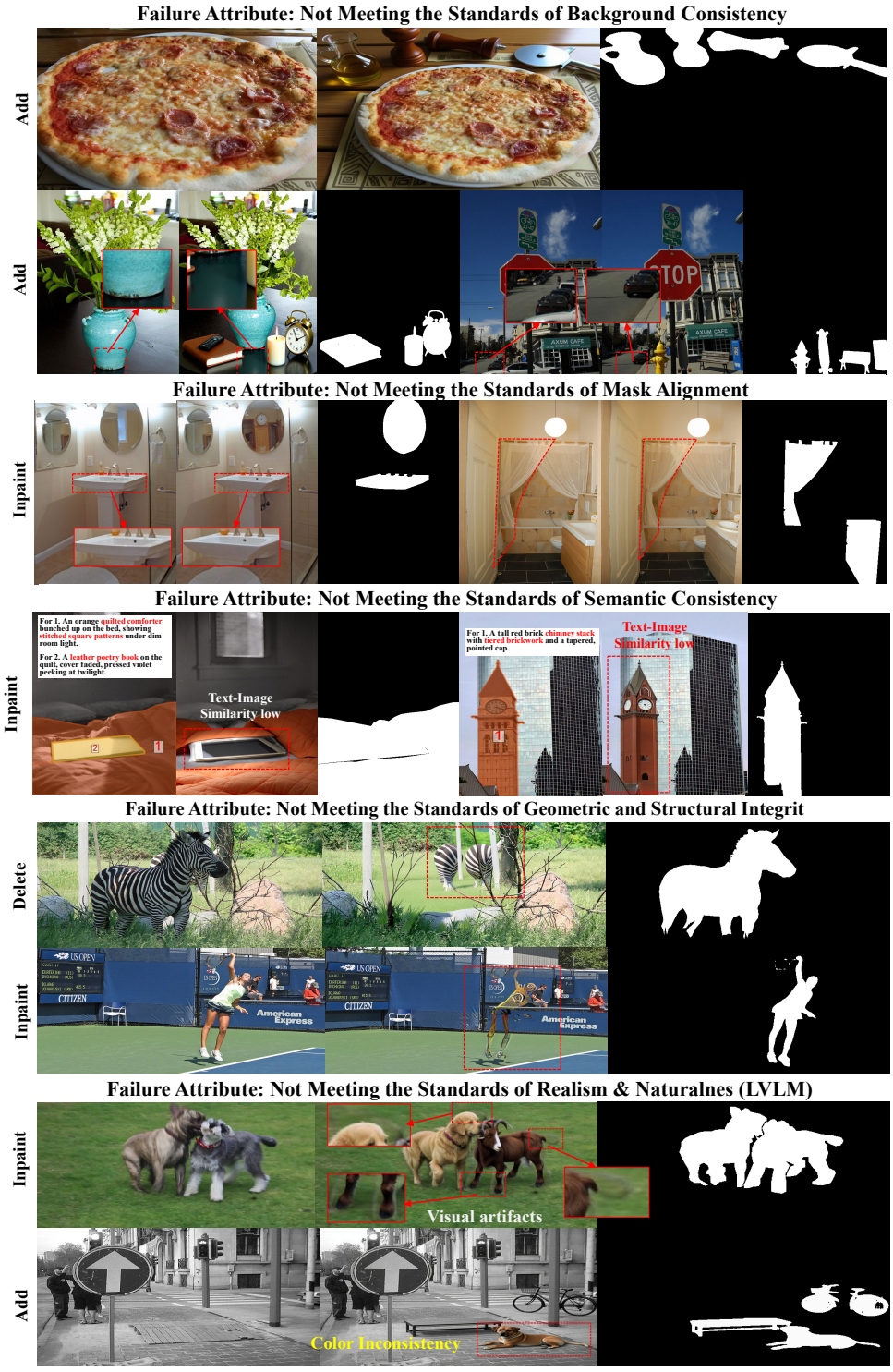

Figure 13: Representative failure samples for different filtering attributes. Each triplet shows the original image, manipulated image, and manipulation mask from left to right.

G.2 Visualization of Representative Rejected Samples During Human Verification

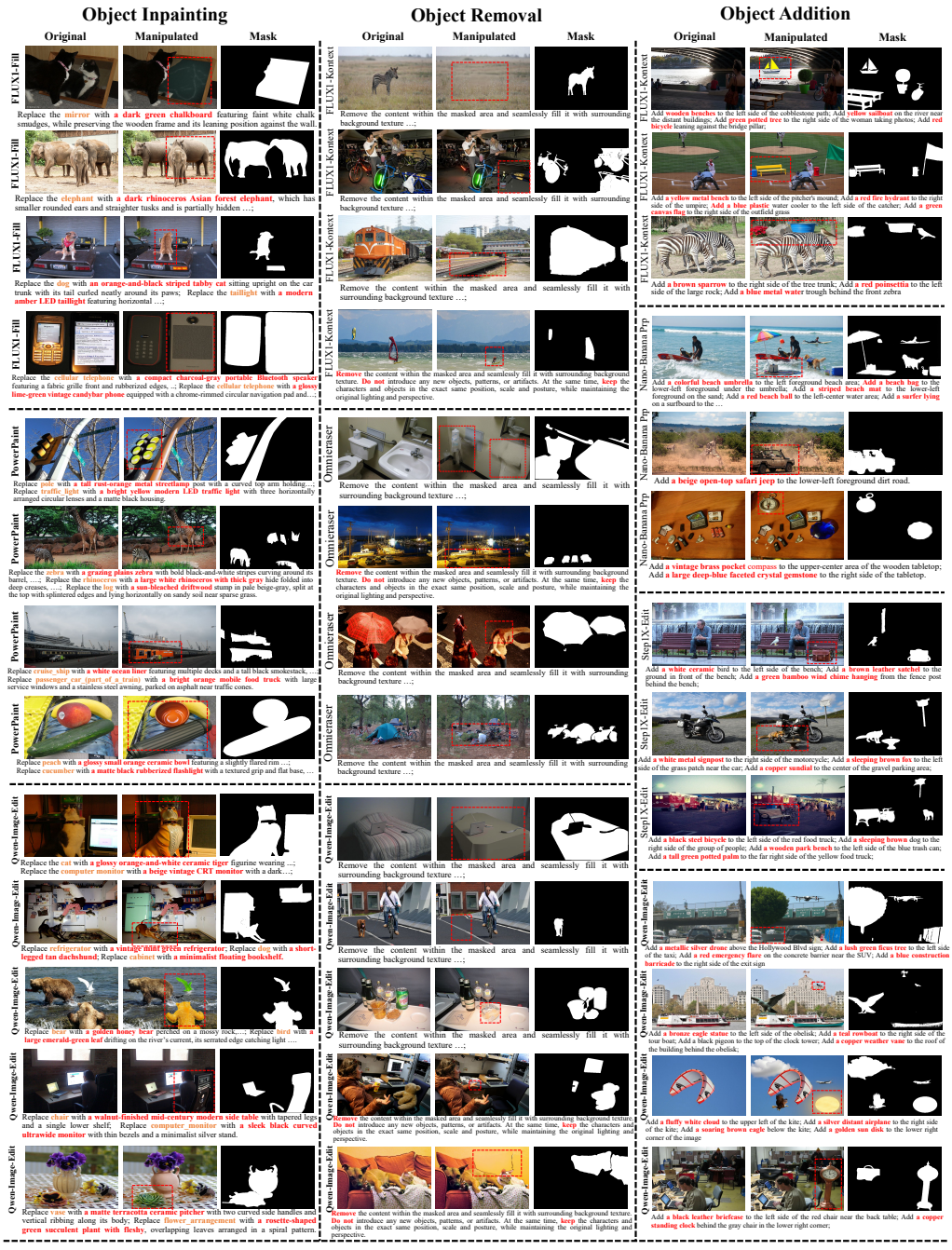

Figure 14: Representative samples rejected during human verification. Each triplet shows the original image, manipulated image, and manipulation mask from left to right. The examples cover object inpainting, object removal, and object addition. Red dashed boxes highlight the main failure regions, including visually implausible edits, obvious generative artifacts, incomplete removal, distorted geometry, and violations of semantic or physical commonsense. These samples are discarded to ensure the realism and annotation reliability of Impostor.

G.3 Visualization of Representative Samples



Figure 15: Representative samples from Impostor. Each triplet shows the original image, manipulated image, and manipulation mask from left to right.