Less Supervision, Better Generalization: Weakly Supervised Fake Region Localization in Diffusion-Edited Images

Junhee Lee
Kyung Hee University
jhlee39@khu.ac.kr

Donghyeon Jeon
Kyung Hee University
amuse_dh@khu.ac.kr

Taeoh Kim
NAVER Cloud
taeoh.kim@navercorp.com

Beomyoung Kim
NAVER Cloud
beomyoung.kim@navercorp.com

MyeongAh Cho*
Kyung Hee University
maycho@khu.ac.kr

Project Page: https://visualsciencelab-khu.github.io/LSBG_project

Abstract

Localizing AI-edited regions is essential for interpretable forensic analysis, but remains challenging due to subtle and spatially distributed artifacts that are misaligned with semantic or object boundaries. Existing approaches rely on pixel-level supervision from controlled editing pipelines, which is difficult to scale and can introduce misleading signals: artifacts frequently extend beyond annotated regions, while out-of-mask pixels are treated as authentic. This limits models' ability to capture transferable evidence and generalize across generators and datasets. To address these issues, we propose **ReGFLoW**, a **Re**construction-**G**uided **F**ake **Lo**calization framework under **W**eak supervision, which is the first weakly supervised approach for diffusion-edited fake region localization. ReGFLoW requires only real/fake labels at the image-level and uses diffusion reconstruction errors as dense spatial guidance to inject them into both feature and score spaces. Furthermore, by artifact-centric multiple instance learning, ReGFLoW utilizes localized diffusion evidence without relying on semantic-affinity or boundary-based pseudo-mask priors. Extensive experiments show competitive cross-generator localization, while ReGFLoW outperforms all evaluated fully supervised baselines when evaluation includes both partially edited and fully synthetic images and in cross-dataset tests, without target-domain adaptation.

1 Introduction

The rapid advancement of generative image models has significantly lowered the barrier to producing highly realistic visual content[1–5]. While these developments enable a wide range of creative applications, they also raise serious concerns regarding the proliferation of synthesized or manipulated images online. As a result, detecting AI-generated images has become a critical issue in digital forensics and security fields[6–9]. However, image-level detection alone provides limited forensic value, as it only indicates whether an image is suspicious without identifying the manipulated regions. To address this limitation, AI-generated image region localization aims to determine authenticity at the regional level[9–17]. This enables more interpretable forensic analysis by providing explicit

*Corresponding author.

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

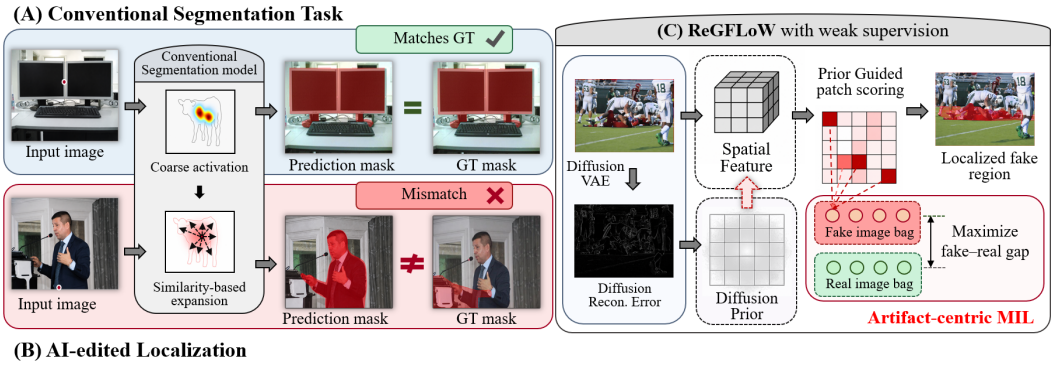

Figure 1: Motivation of ReGFLoW. (A) Conventional localization tasks, such as semantic segmentation and image manipulation localization, often rely on object- or boundary-aligned targets. (B) Diffusion-edited fake regions can be non-semantic and weakly bounded, making similarity-based expansion unreliable. (C) ReGFLoW introduces reconstruction-guided MIL for weakly supervised fake region localization.

evidence of where manipulations have occurred, thereby improving explainability for real-world applications.

Among various generative image models, this paper focuses on localizing manipulated regions in images generated or edited by diffusion-based models. Although this task appears similar to conventional semantic segmentation or image manipulation localization[18–22], it poses fundamentally different challenges that existing approaches cannot adequately address. Semantic segmentation and manipulation localization methods are typically designed for objects or background regions with clear boundaries[18–20, 23–31]. However, in diffusion-based editing, the changes are often less structured—for example, slight background alterations, partial edits to objects, or subtle texture inconsistencies that do not correspond to clear semantic regions.[3–5, 10–14, 16] As a result, the diffusion-edited image localization task requires identifying low-level visual traces rather than relying on high-level semantic cues.

Existing methods for diffusion-edited region localization typically rely on fully supervised learning, using pixel-level manipulation masks generated during the image editing or inpainting process[9, 11–13]. While this approach is simple and effective in controlled benchmark scenarios, there are two reasons to argue that it is unsuitable for realistic diffusion-edited region localization. **First**, obtaining accurate pixel-level ground-truth masks at scale is inherently challenging, like other segmentation tasks. However, aside from this, commercial image editing and generation services mainly take text-based inputs, making it difficult to restrict edits to only the masked region.[32–43] Even if a mask image is provided or the target region is well specified in text, the model may still manipulate regions outside the intended region. These issues make it difficult to obtain reliable edit masks during the image editing process. **Second**, even if the mask is created or obtained by humans, it can lead to unintended behavior due to incorrect supervision. As described above, diffusion-based image manipulation involves latent encoding, denoising, decoding, and blending processes that can leave generation traces beyond the intended editing region defined by the mask.[2–6, 8] However, when imprecise mask-based fully supervised learning is applied without accounting for this, it treats all regions outside the mask as unmanipulated and enforces supervision strictly based on the mask boundaries. Additionally, the misalignment between the mask and the manipulated regions may cause the model to incorrectly learn which regions are manipulated. As a result, the model tends to rely on mask boundaries while ignoring subtle artifacts outside the mask region, leading to spurious correlations. As a result, generalization performance in unseen domains is degraded, as analyzed in detail in Section 2.

Motivated by these issues, we propose the first weakly supervised method for diffusion-edited fake region localization in generic visual content using only image-level real/fake labels. This allows the model to learn from a wider range of edited images without relying on pixel-level masks. It also avoids assuming that diffusion artifacts appear only in the edited regions, which helps improve generalization. However, weakly supervised learning alone poses challenges for precise spatial part localization. Existing weakly supervised approaches address conventional image manipulation [21, 44] or study

diffusion edits in face-centric settings [22]. In our setting, diffusion-edited regions, as shown in Figure 1, are not consistently aligned with semantic object structures, limiting the effectiveness of conventional pseudo-mask refinement and requiring a new task-specific approach for diffusion-edited fake region localization [23–28].

In this paper, we propose **ReGFLoW**, **Re**construction-**G**uided **F**ake **Lo**calization under **W**eak supervision. ReGFLoW learns localization via an *artifact-centric multiple instance learning* (MIL) objective, enabling the model to discover diffusion artifact evidence beyond the intended edit mask. The core idea of ReGFLoW is to overcome the limitations of semantic-affinity or boundary-based spatial priors by leveraging diffusion reconstruction error—previously studied mainly for image-level fake detection—as a prior to identifying fake regions. Specifically, the diffusion reconstruction error serves as dense spatial guidance, injected into both the feature and score spaces. This guidance allows the model to learn from the entire image by capturing generalizable diffusion artifact cues rather than relying solely on source-specific and mask-bound patterns.

Our contributions are summarized as follows:

- We provide a systematic analysis of pixel-level mask supervision for diffusion-edited localization, identifying two generalization-limiting factors: pixel-level masks are often unavailable in realistic data, and mask-based supervised training can suppress transferable diffusion traces by treating out-of-mask regions as belonging to the real class.
- To overcome these generalization limits, we formulate, to the best of our knowledge, the first weakly supervised setting for diffusion-edited fake region localization in generic visual content, where models learn spatial localization from image-level real/fake labels rather than pixel-level manipulation masks.
- We propose **ReGFLoW**, a **Re**construction-**G**uided **F**ake **Lo**calization framework under **W**eak supervision. ReGFLoW repurposes diffusion reconstruction error from image-level fake detection into dense spatial guidance, and combines it with an artifact-centric multiple instance learning objective to localize diffusion artifacts without object- or boundary-based pseudo-mask priors.
- ReGFLoW achieves competitive cross-generator localization and outperforms all evaluated fully supervised baselines when evaluated on both partially edited and fully synthetic images and across datasets, without target-domain adaptation.

2 Rethinking Supervision for Diffusion-Edited Localization

2.1 Pixel Masks Limit Broad-Domain Training

A central goal of diffusion-edited image detection is to build a detector that generalizes across diverse generators (including both models and services) [9–14, 16, 17] and image distributions. Existing research methods train on a single source domain—such as a specific generator or image distribution—and evaluate on unseen domains, which limits cross-domain generalization since the training domain remains fixed. However, real-world systems must be trained across a broader set of domains, as generative models and editing services continue to diversify rapidly [32, 34–36, 38–43]. Therefore, it is important to move beyond such constrained protocols and more deeply study generalization for diffusion-edited image detection.

This goal exposes a fundamental limitation of pixel-level supervision, as localization requires pixel-level manipulation masks. However, many commercial generative and editing systems are text-conditioned and do not provide mask-based interfaces, making such masks difficult to obtain from the final outputs (More details in Sec A). As a result, pixel-level supervision restricts training to a limited set of controlled sources where masks are explicitly available. Weak supervision is therefore particularly suitable for diffusion-edited localization: it not only reduces annotation cost by using image-level labels, but also enables learning from broader and more realistic data distributions (Results in Table 6).

2.2 Pixel Masks Can Provide Misleading Supervision for Diffusion Artifacts

Even when pixel-level masks are available, they can provide misleading supervision for learning diffusion artifacts. While conventional image manipulations are often approximated with variations

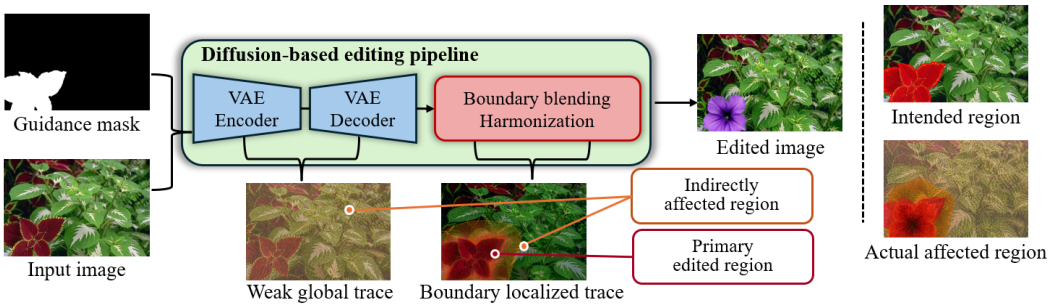

Figure 2: Conceptual illustration of mask incompleteness in diffusion-based editing. The guidance mask indicates the intended edit region, but latent processing and blending can introduce weak global or boundary-localized traces beyond the mask. Thus, the manipulation mask should not be treated as a complete label for all diffusion artifacts.

confined to selected regions [18–20, 29–31], diffusion-based editing can affect a larger spatial region than the intended editing mask [2–5]. As shown in Figure 2, the provided mask indicates the area where editing is intended, but, due to the nature of the diffusion model, the diffusion trace can be extended beyond that mask through mapping to the latent space via the VAE and subsequent blending or harmonization [2, 4, 5, 8]. Therefore, the masks provided alone should not be considered as a correct answer label for the diffusion-edited fake region localization.

These discrepancies directly affect supervised learning, and consequently also affect generalization. Pixel mask supervised learning assigns fake labels to pixels inside the mask and real labels to all pixels outside the mask. If the mask outer region contains weak but transferable diffusion traces, the model is trained to suppress evidence that may arise during the diffusion process itself [6, 8]. Thus, the model relies on stronger but less transferable clues, such as the boundaries of objects confined to masks, specific generative models, or patterns in dataset distributions [9–14]. As a result, if these source-specific clues are absent or weakened in unseen regions, the affected regions may be misclassified as authentic rather than fake.

We observe this tendency in a qualitative motivating example. As shown in Figure 3, the input is an out-of-domain fully synthetic image, where the entire image should be treated as fake. However, the fully supervised baseline predicts only an object-shaped region and suppresses large fake areas as real. This indicates that the model does not simply make boundary errors, but can fail to recognize fake evidence outside the mask-aligned patterns learned during training. This behavior suggests that mask-supervised training can bias the model toward mask-aligned or source-specific cues, rather than transferable diffusion artifacts. When such cues do not hold in unseen domains, fake regions can be missed and classified as real. These observations motivate moving beyond a purely fully supervised mask-prediction paradigm. Rather than serving only as a lower-cost alternative to dense annotation, weak supervision provides a better-aligned formulation for learning generalizable diffusion artifact cues from image-level labels without forcing all out-of-mask regions to be labeled as real [21, 22, 45–47].



Figure 3: Failure case on an out-of-domain fully synthetic image. Although the entire image is fake, the mask-supervised baseline predicts only an object-shaped region and misses large fake areas.

3 Method

3.1 Overview

Figure 4 illustrates the overall pipeline. ReGFLoW learns fake region localization in diffusion-edited images using only image-level real/fake labels. Since diffusion-edited artifacts include both semantic inconsistencies and subtle low-level traces, we use a dual-encoder backbone to capture global semantic cues and local artifact-sensitive features, with intermediate cross-attention exchanging information between the two streams[48, 49].

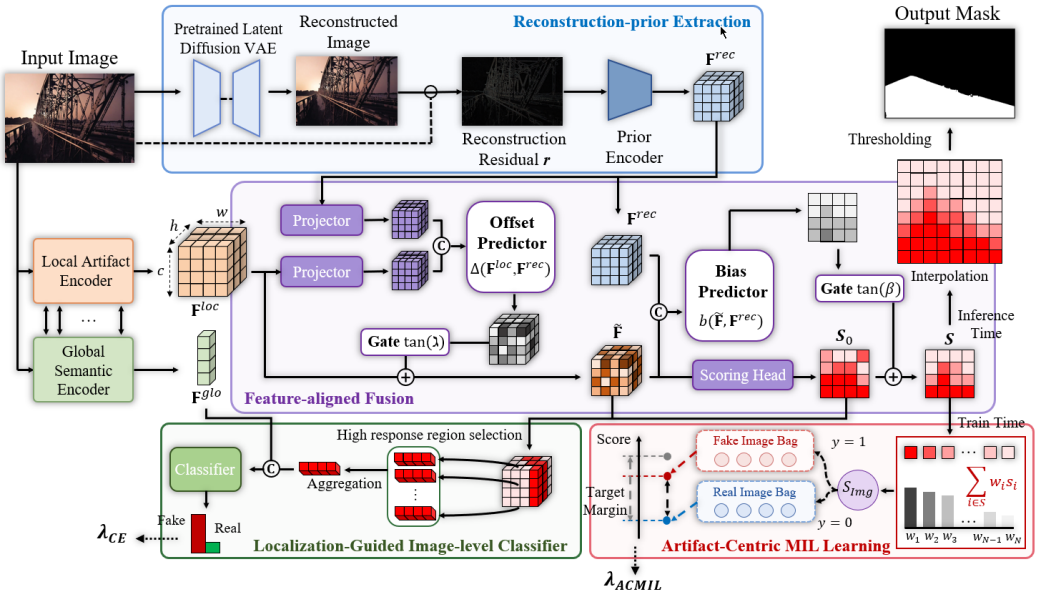

Figure 4: Overview of the proposed ReGFLoW framework. ReGFLoW integrates dual-encoder features and a diffusion reconstruction prior for patch-level fake evidence estimation under weak supervision. The resulting score map is optimized with artifact-centric MIL and used for fake-region localization and image-level classification.

Starting from the resulting dense feature representation, the model proceeds in three steps: (i) a diffusion reconstruction error map is computed and aligned with the local feature space to provide a dense patch-level prior, (ii) the prior is injected into the feature and score spaces through a *reconstruction-guided* patch scoring module, producing a patch score map, and (iii) the score map is trained end-to-end with an artifact-centric multiple instance learning (MIL) objective using only image-level labels[45, 47, 50]. In addition, the localization feature and score map guide the image-level classifier by providing artifact-focused regional evidence.

3.2 Reconstruction-Guided Patch Scoring

Diffusion reconstruction error as a dense prior. Reconstruction error from pretrained diffusion autoencoders has been used as an effective cue for AI-generated image detection [6, 8]. Following this observation, we use the frozen Variational Autoencoder (VAE) of SD1 [2, 51] as a diffusion-aligned reconstructor, since it provides a widely used latent-diffusion reconstruction prior without additional training. Since diffusion-synthesized content has already been projected through this learned latent manifold, it is often reconstructed with a smaller or more regular residual than natural real-image content. We adapt this reconstruction residual as a spatial prior for localization, allowing the network to relate reconstruction patterns to diffusion artifacts at each image region.

Given an input image I, we compute the reconstruction residual using the frozen VAE of SD1:

$$\mathbf{r} = \left| I - \mathcal{D}\big(\mathcal{E}(I)\big) \right|, \tag{1}$$

where $\mathcal{E}$ and $\mathcal{D}$ are the frozen VAE encoder and decoder, respectively, and the absolute difference is taken channel-wise. We use $\mathbf{r}$ as the reconstruction prior.

Feature-aligned fusion. Let $\mathrm{Enc}_{\mathrm{loc}}$ denote the local artifact encoder, which produces a 32×32 patch feature map from the input image. We obtain $\mathbf{F}^{\mathrm{loc}} = \mathrm{AvgPool}_{2\times}(\mathrm{Enc}_{\mathrm{loc}}(I)) \in \mathbb{R}^{B\times C\times 16\times 16}$, where $C = 768$. This compact grid prevents the MIL bag from becoming too large, which improves loss propagation and stabilizes weakly supervised training. Since $\mathbf{r}$ is defined in image space, we align it to the same 16×16 patch grid before fusion. Let $\mathrm{Enc}_{\mathrm{prior}}$ denote the three-layer convolutional Prior Encoder; we compute $\mathbf{F}^{\mathrm{rec}} = \mathrm{AvgPool}_{2\times}(\mathrm{Enc}_{\mathrm{prior}}(\mathbf{r})) \in \mathbb{R}^{B\times C_{\mathrm{rec}}\times 16\times 16}$, where $C_{\mathrm{rec}} = 32$. We inject the reconstruction prior in two complementary ways: (i) Feature-aligned Fusion through residual feature correction and (ii) Bias Prediction through a zero-initialized reconstruction logit bias.

Residual feature fusion. Let P_{loc} and P_{rec} denote two separate convolutional Projectors for $\mathbf{F}^{\mathrm{loc}}$ and $\mathbf{F}^{\mathrm{rec}}$, respectively. The convolutional Offset Predictor G_Δ then predicts the correction offset from their channel-wise concatenation:

$$\Delta(\mathbf{F}^{\mathrm{loc}}, \mathbf{F}^{\mathrm{rec}}) = G_\Delta\big(\big[P_{\mathrm{loc}}(\mathbf{F}^{\mathrm{loc}}); P_{\mathrm{rec}}(\mathbf{F}^{\mathrm{rec}})\big]\big)\,. \tag{2}$$

The offset provides a gated residual correction, so reconstruction cues can adjust local artifact features without replacing them:

$$\widetilde{\mathbf{F}} = \mathbf{F}^{\mathrm{loc}} + \tanh(\gamma) \cdot \Delta(\mathbf{F}^{\mathrm{loc}}, \mathbf{F}^{\mathrm{rec}}), \tag{3}$$

where γ is a learnable scalar initialized to 0.05. The $\tanh(\cdot)$ gate makes the fusion start as a near-identity operation and gradually activates as training progresses, preventing the reconstruction prior from destabilizing early-stage optimization. The resulting feature $\widetilde{\mathbf{F}} \in \mathbb{R}^{B\times C\times 16\times 16}$ is used for patch-level scoring.

Zero-initialized reconstruction logit bias. The fused feature $\widetilde{\mathbf{F}}$ is passed to the one-hidden-layer convolutional Scoring Head, which produces a base patch logit map $\mathbf{S}_0 \in \mathbb{R}^{B\times 16\times 16}$. To further refine the final patch scores, the Bias Predictor b takes the channel-wise concatenation of $\widetilde{\mathbf{F}}$ and $\mathbf{F}^{\mathrm{rec}}$ and predicts a reconstruction logit bias $b(\widetilde{\mathbf{F}}, \mathbf{F}^{\mathrm{rec}}) \in \mathbb{R}^{B\times 16\times 16}$:

$$\mathbf{S} \;=\; \mathbf{S}_0 \;+\; \tanh(\beta) \cdot b(\widetilde{\mathbf{F}},\, \mathbf{F}^{\mathrm{rec}}), \tag{4}$$

where the final 1×1 output convolution of b is *zero-initialized*, and β is a learnable scalar initialized to 0.05. Because the bias branch initially outputs zero, the model starts with $\mathbf{S} = \mathbf{S}_0$, i.e., it behaves identically to a baseline without reconstruction logit bias; the bias term grows only as training provides evidence for its utility.

Together, Eqs. (2)–(4) inject the reconstruction prior into both feature and score spaces, while the gated residual path and zero-initialized bias keep the initial behavior close to the baseline for stable joint optimization.

3.3 Weakly Supervised Training Objective

MIL view of patch scoring. The final logit map $\mathbf{S} \in \mathbb{R}^{B\times 16\times 16}$ is interpreted as a bag of patch instances. For a fake image, at least some patches are expected to contain diffusion-generated traces, whereas a real image should not contain such artifact-positive patches. Let $\mathcal{F}$ and $\mathcal{R}$ denote the fake and real image sets in a batch, respectively. The objective encourages the aggregated artifact score of each fake image to exceed that of each real image, i.e., $\hat{s}_i > \hat{s}_j$ for $i \in \mathcal{F}$ and $j \in \mathcal{R}$, where $\hat{s}$ denotes image-level artifact evidence aggregated from patch logits.

Compared with global pooling, which can dilute small fake regions or encourage broad responses, MIL focuses supervision on the most suspicious patches. This better matches the partial and spatially variable nature of fake regions while still supporting fully synthetic images.

Artifact-centric MIL loss. Let $\{s_{i,n}\}_{n=1}^{N}$ denote the flattened patch logits of image i, where $N = 16 \times 16$. Rather than using hard top-k selection, we use image-wise normalized softmax aggregation to handle large variations in fake-region size[46]:

$$\hat{s}_i = \sum_{n=1}^{N} \alpha_{i,n} s_{i,n}, \qquad \alpha_{i,n} = \frac{\exp(\kappa z_{i,n})}{\sum_{m=1}^{N} \exp(\kappa z_{i,m})}, \qquad z_{i,n} = \frac{s_{i,n} - \mu_i}{\sigma_i + \epsilon}, \tag{5}$$

where $z_{i,n}$ is the image-wise standardized logit, μ_i and σ_i are the mean and standard deviation of the patch logits in image i, and κ controls the sharpness of the aggregation. This aggregation acts as a soft top-k operator, emphasizing patches with strong artifact evidence while remaining fully differentiable. This is well suited to weakly supervised fake region localization in diffusion-edited images, where artifact traces may be weak and spatially diffuse, and the fake region size can vary across samples.

We enforce this ordering with a margin-based pairwise artifact-centric MIL loss:

$$\mathcal{L}_{\mathrm{acmil}} \;=\; \frac{1}{|\mathcal{F}||\mathcal{R}|} \sum_{\substack{i\in\mathcal{F}\\ j\in\mathcal{R}}} \mathrm{softplus}\big(m - \hat{s}_i + \hat{s}_j\big), \tag{6}$$

where $m > 0$ is a margin hyperparameter. This loss penalizes fake–real pairs whose artifact-score gap is smaller than the margin, thereby propagating image-level supervision to the patch score map.

Table 1: P setting (partial edited images only). Pixel-level performance on OpenSDID benchmark. Existing weakly supervised baselines are adapted from conventional manipulation or face-centric deepfake localization. OOD Avg. denotes the average over four cross-domain generators: SD2.1, SDXL, SD3, and Flux.1.

Method	In-domain		Cross-domain								Average			
	SD1.5		SD2.1		SDXL		SD3		Flux.1		Avg.		OOD Avg.	
	F1	IoU	F1	IoU	F1	IoU	F1	IoU	F1	IoU	F1	IoU	F1	IoU
Fully-supervised methods														
PSCC-Net [52]	64.2	54.7	44.8	36.7	26.1	19.7	37.3	29.3	11.6	8.2	36.8	29.7	29.9	23.5
TruFor [19]	71.0	63.4	61.9	54.7	31.9	26.5	38.5	32.2	9.7	7.6	42.6	36.9	35.5	30.3
IML-ViT [18]	73.6	66.5	50.6	44.8	26.0	21.5	28.3	23.6	7.9	6.1	37.3	32.5	28.2	24.0
MaskCLIP [9]	**75.8**	**68.6**	**63.2**	**56.1**	**35.2**	**29.5**	**49.8**	**42.7**	**18.4**	**14.8**	**48.5**	**42.4**	**41.7**	**35.8**
Weakly-supervised methods														
WSCL [21]	25.6	16.7	25.9	17.0	26.0	17.0	25.4	16.5	25.3	16.4	25.6	16.7	25.7	16.7
DOLOS [22]	21.5	13.4	19.1	11.9	19.8	12.2	20.3	12.5	19.2	12.0	20.0	12.4	19.6	12.2
BoxPromptIML [44]	11.5	7.0	10.9	6.5	10.9	6.6	13.8	8.4	10.3	6.2	11.5	6.9	11.5	6.9
Ours	**49.3**	**36.9**	**46.3**	**33.8**	**34.9**	**23.6**	**43.9**	**30.9**	**29.2**	**19.0**	**40.7**	**28.8**	**38.6**	**26.8**

Image-level branch. In addition to localization, ReGFLoW predicts an image-level real/fake label. The selected CLS features from the global semantic encoder are aggregated with a single-layer attention module to obtain a global image embedding. Rather than classifying this embedding alone, we aggregate the top 10% high-response locations of $\widetilde{\mathbf{F}}$ using masked softmax attention from image-wise normalized $\mathbf{S}$. The resulting artifact-focused context is concatenated with the global image embedding and classified by a two-layer MLP with standard cross-entropy $\mathcal{L}_{\mathrm{ce}}$.

Overall objective. The full training objective is

$$\mathcal{L} = \lambda_{\mathrm{acmil}}\,\mathcal{L}_{\mathrm{acmil}} + \lambda_{\mathrm{ce}}\,\mathcal{L}_{\mathrm{ce}} + \lambda_{\mathrm{real}}\,\mathcal{L}_{\mathrm{real}} + \lambda_{\mathrm{smooth}}\,\mathcal{L}_{\mathrm{smooth}}. \tag{7}$$

Here, $\mathcal{L}_{\mathrm{real}}$ is a real-image hard-negative suppression term that discourages high fake scores on real-image patches, and $\mathcal{L}_{\mathrm{smooth}}$ is an edge-aware smoothness term that promotes spatially coherent localization while preserving image boundaries. Their detailed formulations are provided in Appendix B.

Inference and calibration. At inference, $\mathbf{S}$ is bilinearly upsampled to the input resolution. Before thresholding, we apply image-wise adaptive calibration to account for sample-dependent score distributions under weak supervision. Each score map is normalized by its own mean and standard deviation, with the offset adjusted according to its mean uncalibrated positive response. The calibrated probability map is then thresholded to obtain the final fake-region mask. Detailed formulations are provided in Appendix B.6.

4 Experiments

4.1 Experimental Setup

We conduct experiments on OpenSDID, a benchmark for diffusion-generated and diffusion-edited image detection and localization [9]. Following the OpenSDID protocol, we train on the SD1.5 split and evaluate on SD1.5 as the in-domain setting and on SD2.1, SDXL, SD3, and Flux.1 as cross-domain settings. Although OpenSDID provides both image-level real/fake labels and pixel-level masks, ReGFLoW uses only image-level labels during training. Pixel-level masks are reserved solely for evaluation. As weakly supervised baselines, we adapt WSCL [21] and BoxPromptIML [44] from conventional image manipulation localization, and DOLOS [22] from face-centric diffusion-edited localization. We additionally evaluate cross-dataset localization on COCO-GLIDE [53] and DOLOS [22], without target-domain adaptation. Architectural and training details, including the backbone configuration, are provided in Appendix C.

4.2 Pixel-Level Localization Evaluation

Table 1evaluates localization on partially edited fake images. The adapted weakly supervised baselines yield poor localization even in-domain, suggesting that their formulations do not readily extend to

Table 2: P+F setting (partial edited and fully synthetic images). Pixel-level F1 evaluated on OpenSDID. OOD Avg. denotes the average over four cross-domain generators.

Method	In-domain	Cross-domain				Average	
	SD1.5	SD2.1	SDXL	SD3	Flux.1	Avg.	OOD Avg.
Fully-supervised methods							
TruFor [19]	28.8	20.1	15.9	16.4	**8.4**	17.9	15.2
IML-ViT [18]	30.2	24.7	17.6	13.5	7.7	18.7	15.9
MaskCLIP [9]	**95.1**	**73.2**	**20.8**	**17.7**	4.8	**42.3**	**29.1**
Weakly-supervised methods							
Ours	**35.6**	**33.4**	**30.1**	**32.2**	**29.4**	**32.2**	**31.3**

Table 3: Cross-dataset pixel-level F1 on COCO-GLIDE [53] and DOLOS [22]. P denotes partially edited images, while P+F includes both partially edited and fully synthetic images. All methods are evaluated without target-domain adaptation.

Supervision	Method	COCO-GLIDE	DOLOS	
		P	P	P+F
Fully-supervised	TruFor [19]	**25.17**	8.21	12.70
	IML-ViT [18]	15.23	1.16	1.74
	MaskCLIP [9]	8.34	**13.35**	**14.65**
Weakly-supervised	**Ours**	**42.01**	**21.09**	**21.19**

this task. Fully supervised baselines achieve strong in-domain scores on SD1.5 but degrade sharply on unseen generators. This is consistent with our analysis in Section 2 that pixel-mask supervision can encourage reliance on mask-confined, source-specific cues that transfer poorly across generators. In contrast, ReGFLoW, despite using only image-level labels, achieves competitive OOD Avg F1 (38.6), surpassing most fully supervised baselines with markedly more stable cross-domain behavior. The benefit is most pronounced on Flux.1 (29.2 vs. 18.4), the generator most distant from SD1.5, demonstrating that reconstruction-guided weak supervision effectively captures transferable diffusion traces where mask-bounded supervision fails.

Table 2 evaluates the more realistic setting where both training and evaluation contain partially edited and fully synthetic fake images. Here the limitation of mask-based supervision becomes far more severe: MaskCLIP retains high in-domain F1 (95.1) but collapses across all unseen generators, yielding only 29.1 OOD Avg. ReGFLoW achieves the highest OOD Avg. F1 of 31.3, outperforming all evaluated fully supervised baselines while using no pixel-level supervision or target-domain adaptation. This suggests stronger generalization not only across image distributions, but also across fake-image types, as localization in this setting must handle both localized edits and fully generated images.

Beyond the cross-generator evaluation within OpenSDID, we further assess cross-dataset generalization under broader distribution shifts using COCO-GLIDE [19, 53] and DOLOS [22]. Unlike the latent-space generative pipelines used in OpenSDID, GLIDE performs diffusion directly in image space, introducing a distinct generation mechanism, while DOLOS targets face-centric deepfakes, resulting in a substantial shift in content domain. As shown in Table 3, ReGFLoW achieves the highest F1 across all three settings, outperforming every evaluated fully supervised baseline without target-domain adaptation.

4.3 Image-Level Detection Evaluation

Table 4 reports image-level detection performance on OpenSDID. Although ReGFLoW is primarily designed for localization, its image-level branch directly leverages localization information by fusing high-response regions from the patch score map with the global CLS embedding before classification, providing the detector with artifact-focused regional evidence rather than relying solely on a whole-image representation. As a result, ReGFLoW achieves the highest OOD Avg, outperforming not only image-only baselines but also methods that additionally use pixel-mask supervision. Notably, image-only baselines generally surpass methods jointly trained with pixel-mask supervision in OOD Avg, suggesting that mask annotations bias the model toward source-specific spatial patterns that hinder

Table 4: Image-level detection performance on OpenSDID. OOD Avg denotes the average over four cross-domain generators: SD2.1, SDXL, SD3, and Flux.1.

Approach	Method	SD1.5		SD2.1		SDXL		SD3		Flux.1		Avg		OOD Avg	
		F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC
Image only	CNNDet[54]	84.60	85.04	71.56	75.94	59.70	68.72	56.27	67.08	35.72	57.57	61.57	70.87	55.81	67.33
	GramNet[55]	80.51	80.35	74.01	76.66	65.28	70.76	64.35	70.29	52.00	63.37	67.23	72.29	63.91	70.27
	FreqNet[56]	75.88	77.70	60.97	68.37	53.15	64.02	53.50	64.37	38.47	57.08	56.39	66.31	51.52	63.46
	NPR[57]	79.41	79.28	81.67	81.84	72.12	74.28	73.43	75.47	**67.62**	**71.36**	74.85	76.45	73.71	75.74
Image &Pixel	TruFor[19]	90.12	**97.73**	35.93	55.62	58.04	66.41	59.73	67.51	49.12	61.62	58.59	69.78	50.70	62.79
	IML-ViT[18]	**94.47**	75.73	69.70	61.19	40.98	49.95	44.69	51.25	18.20	43.62	53.61	56.35	43.39	51.50
	MaskCLIP[9]	91.28	91.47	83.36	85.28	72.90	77.98	67.79	74.95	53.08	67.12	73.68	79.36	69.28	76.33
	Ours	93.02	93.08	**91.34**	**91.73**	**79.89**	**82.63**	**76.87**	**80.62**	56.78	68.76	**79.58**	**83.36**	**76.22**	**80.93**

cross-domain transfer even at the image level. ReGFLoW circumvents this trade-off by deriving regional evidence from weakly supervised, reconstruction-guided localization, thereby benefiting from spatial information without inheriting the generalization cost of pixel-mask supervision.

4.4 Ablation Studies

Table 5 ablates the main modules shown in Figure 4. *Feature-aligned fusion* uses the reconstruction prior to modulate local artifact features through a gated offset, while the *bias predictor* adds reconstruction-guided evidence directly to the patch score map. *Artifact-centric MIL* denotes our adaptive top-k aggregation with fake-real pairwise ranking supervision.

Table 5: Ablation studies of ReGFLoW modules. The module names follow Figure 4.

Method	Bias Predictor	Feature-aligned Fusion	Artifact-Centric MIL	Pixel F1
ReGFLoW	✓	✓	✓	**49.38**
w/o Bias Predictor ×		✓	✓	46.06
w/o Reconstruction Prior	×	×	✓	38.08
Score-Pooling BCE	×	×	×	32.34

Removing the bias predictor weakens the score-level use of the reconstruction prior, and removing the reconstruction prior entirely causes a larger drop. Replacing artifact-centric MIL with image-level BCE on a pooled score further degrades performance, showing that both reconstruction-guided scoring and artifact-centric weak supervision are important for localization.

4.5 Scaling to Broader Training Domains

A practical motivation for weak supervision, as discussed in Section 2, is that it enables training on broader and more diverse data sources without requiring pixel-level masks. We test whether ReGFLoW can convert this flexibility into improved cross-domain localization. Using OpenSDID, we extend the SD1.5 training set with one additional source domain while keeping the total number of training images fixed, isolating the effect of domain diversity from data scale by uniformly subsampling each source. As shown in Table 6, adding any single source domain consistently improves average pixel-level F1 on unseen domains, with the largest gains observed on the most distant generators where SD1.5-only training provides the weakest baseline. These results show that weak supervision is not merely a lower-cost alternative to dense annotation, but a practical means of scaling diffusion-edited localization to the broader data distributions that pixel-mask supervision cannot accommodate.

Table 6: Effect of source-domain diversity on ReGFLoW's cross-domain localization. Each row adds one source domain to SD1.5 while keeping the total training size fixed.

Train	SD2	SD3	SDXL	FLUX	Avg. Δ
SD1.5	40.96	38.09	29.36	24.12	–
+ SD2	–	42.34 (+4.25)	33.73 (+4.37)	28.71 (+4.60)	+4.40
+ SD3	42.75 (+1.79)	–	34.48 (+5.12)	30.96 (+6.84)	+4.58
+ SDXL	40.88 (-0.08)	41.84 (+3.75)	–	31.00 (+6.88)	+3.52
+ FLUX	41.57 (+0.61)	43.82 (+5.73)	34.47 (+5.11)	–	+3.82

4.6 Robustness to Image Degradations

We further compare ReGFLoW with additional fully supervised baselines [30, 31, 58, 59] under Gaussian blur and JPEG compression, following the degradation settings reported in [9]. As shown in

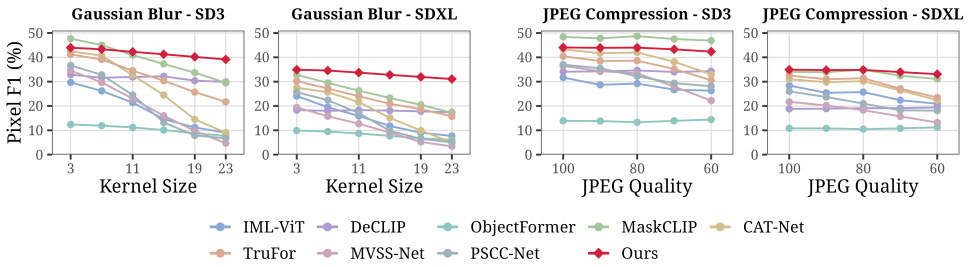

Figure 5: Robustness to Gaussian blur and JPEG compression on SD3 and SDXL. Baseline curves are digitized from the robustness results reported in OpenSDI [9], and ReGFLoW is evaluated under the same degradation settings.

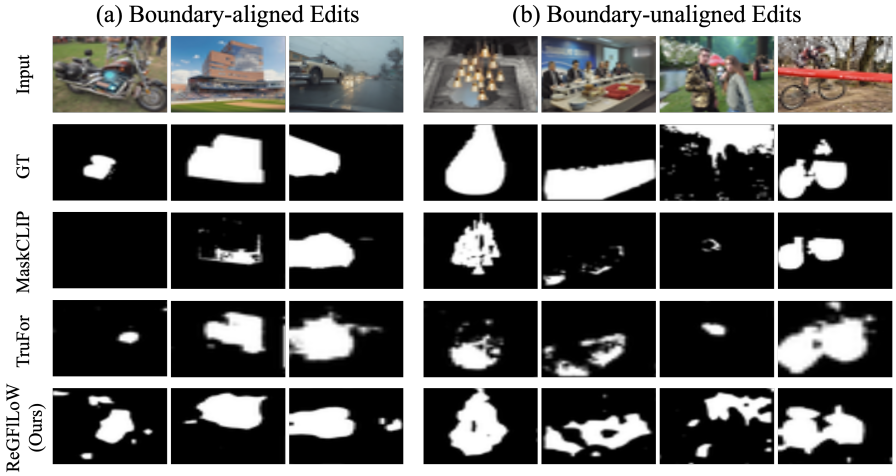

Figure 6: Qualitative comparison on out-of-domain fake-region localization. MaskCLIP [9] and TruFor [19] are fully supervised, whereas Ours is weakly supervised. The examples include (a) boundary-aligned edits and (b) boundary-unaligned edits, where boundaries provide limited guidance.

Figure 5, ReGFLoW degrades gradually under increasingly strong Gaussian blur and remains stable under JPEG compression across both SD3 and SDXL. Under severe blur, ReGFLoW outperforms the compared baselines on both generators, while remaining highly competitive under JPEG compression, indicating that its localization is not solely dependent on fragile high-frequency cues.

4.7 Qualitative Analysis

Figure 6 presents qualitative localization results on out-of-domain examples. We include (a) boundary-aligned edits, similar to conventional segmentation-style data, as well as (b) boundary-unaligned cases involving background edits, partial changes inside an object, and multiple manipulated objects. We compare ReGFLoW with fully supervised baselines, MaskCLIP and TruFor. These examples show that when object boundaries provide limited guidance, ReGFLoW better localizes artifact evidence itself rather than simply following semantic or object-aligned regions.

5 Conclusion

In this paper, we presented ReGFLoW, a weakly-supervised framework for AI-edited fake region localization. To address the limitations of fully-supervised methods—namely, the difficulty of collecting scalable annotations and the risk of misleading supervision signals—ReGFLoW learns from image-level real/fake labels and leverages diffusion reconstruction error as dense spatial guidance. ReGFLoW achieves competitive cross-generator localization and surpasses all evaluated fully supervised baselines when both partially edited and fully synthetic images are considered, as well as in cross-dataset evaluation, without target-domain adaptation. These results suggest that weakly supervised learning is a practical formulation for learning generalizable features for AI-edited fake region localization as a scalable alternative to mask-supervised frameworks.

Acknowledgments and Disclosure of Funding

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government(MSIT)(RS-2024-00456589) and Korea Planning & Evaluation Institute of Industrial Technology(KEIT) grant funded by the Korea government(MOTIE) (No. RS2024-00442120, Development of AI technology capable of robustly recognizing abnormal and dangerous situations and behaviors during night and bad weather conditions)

References

[1] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020.

[2] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[3] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022.

[4] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[5] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. RePaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[6] 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*, 2023.

[7] Christos Koutlis and Symeon Papadopoulos. Leveraging representations from intermediate encoder-blocks for synthetic image detection. In *European Conference on Computer Vision*, 2024.

[8] Jonas Ricker, Denis Lukovnikov, and Asja Fischer. AEROBLADE: Training-free detection of latent diffusion images using autoencoder reconstruction error. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9130–9140, 2024.

[9] 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*, 2025.

[10] Hannes Mareen, Dimitrios Karageorgiou, Glenn Van Wallendael, Peter Lambert, and Symeon Papadopoulos. TGIF: Text-guided inpainting forgery dataset. In *IEEE International Workshop on Information Forensics and Security*, 2024.

[11] Valentina Bazyleva, Nicolo Bonettini, and Gaurav Bharaj. X-Edit: Detecting and localizing edits in images altered by text-guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2025.

[12] Rui Zhang, Hongxia Wang, Hangqing Liu, Yang Zhou, and Qiang Zeng. DEAL-300K: Diffusion-based editing area localization with a 300k-scale dataset and frequency-prompted baseline, 2025.

[13] 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 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

[14] Hannes Mareen, Dimitrios Karageorgiou, Paschalis Giakoumoglou, Peter Lambert, Symeon Papadopoulos, and Glenn Van Wallendael. TGIF2: Extended text-guided inpainting forgery dataset & benchmark, 2026.

[15] 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*, 2026.

[16] Alex Costanzino, Woody Bayliss, Juil Sock, Marc Gorriz Blanch, Danijela Horak, Ivan Laptev, Philip Torr, and Fabio Pizzati. Towards reliable identification of diffusion-based image manipulations. In *Advances in Neural Information Processing Systems*, 2025.

[17] Hengrui Kang, Siwei Wen, Zichen Wen, Junyan Ye, Weijia Li, Peilin Feng, Baichuan Zhou, Bin Wang, Dahua Lin, Linfeng Zhang, and Conghui He. LEGION: Learning to ground and explain for synthetic image detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

[18] Xiaochen Ma, Bo Du, Zhuohang Jiang, Xia Du, Ahmed Y. Al Hammadi, and Jizhe Zhou. IML-ViT: Benchmarking image manipulation localization by vision transformer, 2023.

[19] 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*, 2023.

[20] 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*, 2023.

[21] Yuanhao Zhai, Tianyu Luan, David Doermann, and Junsong Yuan. Towards generic image manipulation detection with weakly-supervised self-consistency learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

[22] Dragos-Constantin Tantaru, Elisabeta Oneata, and Dan Oneata. Weakly-supervised deepfake localization in diffusion-generated images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024.

[23] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[24] Alexander Kolesnikov and Christoph H. Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In *European Conference on Computer Vision*, 2016.

[25] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[26] Jiwoon Ahn, Sunghyun Cho, and Suha Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[27] Yude Wang, Jie Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[28] Zhaozheng Chen, Tan Wang, Xiongwei Wu, Xian-Sheng Hua, Hanwang Zhang, and Qianru Sun. Class re-activation maps for weakly-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[29] Yue Wu, Wael AbdAlmageed, and Premkumar Natarajan. ManTra-Net: Manipulation tracing network for detection and localization of image forgeries with anomalous features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[30] Myung-Joon Kwon, In-Jae Yu, Seung-Hun Nam, and Heung-Kyu Lee. CAT-Net: Compression artifact tracing network for detection and localization of image splicing. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021.

[31] Xinru Chen, Chengbo Dong, Jiaqi Ji, Juan Cao, and Xirong Li. Image manipulation detection by multi-view multi-scale supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

[32] OpenAI. Images in ChatGPT. `https://help.openai.com/en/articles/11084440-images-in-chatgpt`, . Accessed: 2026-05-06.

[33] Adobe. Using masks in Adobe Firefly apis for image manipulation. `https://developer.adobe.com/firefly-services/docs/firefly-api/guides/concepts/masking/`. Accessed: 2026-05-06.

[34] Midjourney. Editor. https://docs.midjourney.com/hc/en-us/articles/32764383466893-Editor. Accessed: 2026-05-06.

[35] Canva. Use magic edit to add, replace, and modify photos. https://www.canva.com/help/using-magic-edit/. Accessed: 2026-05-06.

[36] CapCut. Quick and simple AI image replacer. https://www.capcut.com/tools/ai-replace. Accessed: 2026-05-06.

[37] Google Cloud. Edit image without a mask using Imagen. https://docs.cloud.google.com/vertex-ai/generative-ai/docs/samples/generativeaionvertexai-imagen-edit-image-mask-free. Accessed: 2026-05-06.

[38] Google AI for Developers. Nano banana image generation. https://ai.google.dev/gemini-api/docs/image-generation. Accessed: 2026-05-06.

[39] Black Forest Labs. FLUX.2 image editing. https://bfl.mintlify.app/guides/prompting_editing_overview. Accessed: 2026-05-06.

[40] ByteDance Seed. Seedream 4.0. https://seed.bytedance.com/en/seedream4_0. Accessed: 2026-05-06.

[41] Qwen. Qwen-Image-Edit: Image editing with higher quality and efficiency. https://qwen.ai/blog?id=qwen-image-edit, 2025. Accessed: 2026-05-06.

[42] Tencent Hunyuan. HunyuanImage-3.0: A powerful native multimodal model for image generation. https://github.com/Tencent-Hunyuan/HunyuanImage-3.0. Accessed: 2026-05-06.

[43] Shitao Xiao, Yueze Wang, Junjie Zhou, Huaying Yuan, Xingrun Xing, Ruiran Yan, Chaofan Li, Shuting Wang, Tiejun Huang, and Zheng Liu. OmniGen: Unified image generation. *arXiv preprint arXiv:2409.11340*, 2024.

[44] Zhiqing Guo, Dongdong Xi, Songlin Li, and Gaobo Yang. From passive perception to active memory: A weakly supervised image manipulation localization framework driven by coarse-grained annotations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 336–344, 2026. doi: 10.1609/aaai.v40i1.36995.

[45] Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1–2):31–71, 1997.

[46] Maximilian Ilse, Jakub M. Tomczak, and Max Welling. Attention-based deep multiple instance learning. In *International Conference on Machine Learning*, 2018.

[47] Pedro O. Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[48] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021.

[49] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[50] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6479–6488, 2018.

[51] CompVis. Stable diffusion v1-1 model card. https://huggingface.co/CompVis/stable-diffusion-v1-1, 2022. Accessed: 2026-05-06.

[52] 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, 2022.

[53] Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pages 16784–16804, 2022.

[54] 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*, pages 8695–8704, 2020.

[55] Zhengzhe Liu, Xiaojuan Qi, and Philip H. S. Torr. Global texture enhancement for fake face detection in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8060–8069, 2020.

[56] 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*, pages 3247–3258. PMLR, 2020.

[57] Chuangchuang Tan, Huan Liu, 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*, pages 28130–28139, 2024.

[58] Stefan Smeu, Elisabeta Oneata, and Dan Oneata. DeCLIP: Decoding CLIP representations for deepfake localization. In *Proceedings of the Winter Conference on Applications of Computer Vision*, pages 149–159, 2025.

[59] 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*, pages 2364–2373, 2022.

[60] Hugging Face Diffusers. Inpainting. `https://huggingface.co/docs/diffusers/using-diffusers/inpaint`. Accessed: 2026-05-06.

[61] Black Forest Labs. Introducing FLUX.1 tools. `https://bfl.ai/flux-1-tools/`, 2024. Accessed: 2026-05-06.

[62] OpenAI. Image generation: Edits and inpainting. `https://developers.openai.com/api/docs/guides/image-generation`, . Accessed: 2026-05-06.

[63] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *International Conference on Learning Representations*, 2024. URL `https://proceedings.iclr.cc/paper_files/paper/2024/hash/081b08068e4733ae3e7ad019fe8d172f-Abstract-Conference.html`.

[64] Xiaochen Ma, Xuekang Zhu, Lei Su, Bo Du, Zhuohang Jiang, Bingkui Tong, Zeyu Lei, Xinyu Yang, Chi-Man Pun, Jiancheng Lv, and Ji-Zhe Zhou. Imdl-benco: A comprehensive benchmark and codebase for image manipulation detection & localization. *Advances in Neural Information Processing Systems*, 37: 134591–134613, 2025.

Appendix Overview

Appendix A discusses mask availability and the limitations of before–after differencing. Appendices B and C detail the training objective, inference calibration, and implementation. Appendix D presents controlled supervision comparisons, reconstruction-prior analyses, input robustness, and threshold sensitivity. Appendices E–H provide additional qualitative results, limitations, related work, and asset licenses.

A Why Pixel-Level Masks Do Not Scale to Broad Real-World Diffusion Editing

As discussed in Sec. 2.1, pixel-level masks are available in controlled, mask-conditioned pipelines but are not a general supervision source for real-world diffusion editing. Hosted, conversational, and consumer-facing systems often expose only the input, instruction, and edited output, not the underlying generator, editing operation, or full spatial support of generated artifacts.

A.1 Mask Availability Across Current Image Editing Systems

A *conditioning mask* guides an inpainting model; a *UI-level selection* specifies a brush, click, or region in an editing interface; and a *fake-region supervision mask* labels the spatial support of generated or edited artifacts. The first two describe editing controls, not necessarily the complete artifact support after latent encoding, denoising, decoding, blending, and harmonization.

Table 7 summarizes three workflow categories. Stable Diffusion/SDXL inpainting, Kandinsky inpainting, and FLUX.1 Fill explicitly use masks, making them convenient for controlled localization benchmarks but potentially biasing training toward mask-conditioned sources [60, 61]. Consumer-facing tools such as ChatGPT Images, Adobe Firefly, Midjourney Editor, Canva Magic Edit, and CapCut expose selection-based controls [32–36]. These controls may not be preserved when images are shared or collected, and some APIs operate with or without masks [37, 62]. Instruction- or reference-based models, including Gemini/Nano Banana, FLUX.2, Seedream, Qwen-Image-Edit, Hunyuan Image, and OmniGen, further separate editing from explicit spatial annotation [38–43]. Their inputs and outputs may be available without reliable dense artifact labels.

Table 7: Mask availability across representative editing workflows. Edit controls do not necessarily provide reliable pixel-level artifact labels.

Workflow category	**Examples**	**Implication for localization supervision**
Mask-conditioned inpainting pipelines	Stable Diffusion/SDXL inpainting, Kandinsky inpainting, FLUX.1 Fill [60, 61]	Intended edit masks support controlled benchmarks but may not cover the full artifact support.
Consumer-facing tools with UI-level selection	ChatGPT Images, Adobe Firefly, Midjourney Editor, Canva Magic Edit, CapCut [32–36]	Selections may be lost after sharing and do not reveal the full artifact support.
Instruction- or reference-based editing models	Gemini/Nano Banana, FLUX.2, Seedream, Qwen-Image-Edit, Hunyuan Image, OmniGen [38–43]	Inputs, instructions, references, and outputs may be available without dense artifact labels.

Image-level real/fake labels therefore support broader training sources than pixel-level masks, without assuming that all fake evidence is confined to a known edit region.

A.2 Why Before–After Image Differences Do Not Provide Reliable Masks

We examine before–after differencing using Gemini, Qwen-Image, and FLUX.2. Given an original image I and an instruction, we generate an edited image $\widetilde{I}$, resize it to the original resolution, and compute RGB and grayscale differences:

$$\mathbf{D}_{\mathrm{RGB}} = \left| I - \widetilde{I} \right|, \qquad \mathbf{D}_{\mathrm{gray}} = \frac{1}{3} \sum_{c=1}^{3} \left| I^{(c)} - \widetilde{I}^{(c)} \right|. \tag{8}$$

We use two instructions targeting object replacement and a less localized scene property, respectively:

- **Object-level edit:** "Replace the red bus in the image with a yellow taxi, keeping the scene realistic and consistent."
- **Scene-level edit:** "Change the scene to daytime by adjusting lighting, shadows, and sky naturally."

Figure 7 shows that even object-level edits produce differences outside the intended region through changes in viewpoint, object scale, boundaries, geometry, texture, color, illumination, shadows, reflections, and background harmonization. These are rendering changes rather than simply annotation noise. Scene-level edits can affect the sky, road, objects, contrast, and color tone simultaneously, with no single closed target region. Thresholded difference masks consequently depend on alignment, exposure, color normalization, and the chosen threshold.

A difference map measures pixel disagreement, not artifact support: it can respond to geometric or photometric shifts in unchanged semantic regions while missing artifacts with weak RGB differences. Global edits, such as lighting changes, background harmonization, or style modification, further undermine the notion of a single binary edit mask. Thus, before–after differencing does not resolve the mask-availability problem and motivates learning artifact evidence from image-level labels.

B Additional Details of the Training Objective and Inference Calibration

All localization losses operate on the MIL logit map $\mathbf{S} \in \mathbb{R}^{B\times h\times w}$ before upsampling, with $h = w = 16$ by default. Let $N = hw$ and $\mathbf{s}_i \in \mathbb{R}^N$ denote the number of patches and flattened logits of image i. Labels $y_i = 1$ and $y_i = 0$ denote fake and real images, whose minibatch index sets are $\mathcal{F}$ and $\mathcal{R}$. The full objective is

$$\begin{aligned}\mathcal{L} = \lambda_{\text{ce}}\,\mathcal{L}_{\text{ce}} + \lambda_{\text{acmil}}\,\mathcal{L}_{\text{acmil}} + \lambda_{\text{real}}\,\mathcal{L}_{\text{real}} + \lambda_{\text{smooth}}\,\mathcal{L}_{\text{smooth}} \\ + \lambda_{\text{sparse}}\,\mathcal{L}_{\text{sparse}} + \lambda_{\text{con}}\,\mathcal{L}_{\text{con}} + \lambda_{\text{peak}}\,\mathcal{L}_{\text{peak}}.\end{aligned} \tag{9}$$

The default weights are $\lambda_{\text{ce}} = 1.0$, $\lambda_{\text{acmil}} = 1.0$, $\lambda_{\text{real}} = 0.05$, $\lambda_{\text{smooth}} = 0.03$, $\lambda_{\text{sparse}} = 3\times 10^{-3}$, $\lambda_{\text{con}} = 0.02$, and $\lambda_{\text{peak}} = 0$. The optional peak-separation term is disabled by default.

B.1 Real-Image Hard-Negative Suppression

MIL ranking can leave a few high-scoring patches in real images, causing false positives. We suppress these responses using the mean of the top fraction ρ_{real} of their logits:

$$T_\rho(\mathbf{s}_i) = \frac{1}{k}\sum_{n\in \text{TopK}(\mathbf{s}_i,k)} s_{i,n}, \qquad k = \lceil \rho N \rceil\,. \tag{10}$$

$$\mathcal{L}_{\text{real}} = \frac{1}{|\mathcal{R}|}\sum_{i\in\mathcal{R}} \text{softplus}\left(T_{\rho_{\text{real}}}(\mathbf{s}_i)\right). \tag{11}$$

This penalizes the most suspicious real-image patches; we use $\rho_{\text{real}} = 0.10$.

B.2 Edge-Aware Smoothness Regularization

To reduce noise without direct boundary supervision, we apply edge-aware smoothness to fake images in two spatial dimensions rather than to flattened patch scores. Let

$$\mathbf{P}_i = \sigma(\mathbf{S}_i) \in [0,1]^{h\times w} \tag{12}$$

be the patch-level fake probability map. We downsample the input image to the same resolution and average its channels to obtain $\mathbf{G}_i \in \mathbb{R}^{h\times w}$. Horizontal and vertical differences are

$$\nabla_x \mathbf{P}_{i,u,v} = \mathbf{P}_{i,u,v+1} - \mathbf{P}_{i,u,v}, \tag{13}$$

$$\nabla_y \mathbf{P}_{i,u,v} = \mathbf{P}_{i,u+1,v} - \mathbf{P}_{i,u,v}, \tag{14}$$

with edge-aware weights

$$\mathbf{W}^x_{i,u,v} = \exp\left(-\gamma\,|\mathbf{G}_{i,u,v+1} - \mathbf{G}_{i,u,v}|\right), \tag{15}$$

$$\mathbf{W}^y_{i,u,v} = \exp\left(-\gamma\,|\mathbf{G}_{i,u+1,v} - \mathbf{G}_{i,u,v}|\right). \tag{16}$$



(a) Gemini



(b) Qwen-Image



(c) FLUX.2

Figure 7: Before–after differencing across three editing systems. Each panel shows edited outputs and grayscale differences from the original image. Geometric, photometric, and harmonization changes make these maps unreliable as pixel-level artifact labels.

The resulting loss is

$$\mathcal{L}_{\text{smooth}} = \frac{1}{|\mathcal{F}|} \sum_{i \in \mathcal{F}} \left[\frac{1}{h(w-1)} \sum_{u,v} \mathbf{W}^{x}_{i,u,v} \left(\nabla_x \mathbf{P}_{i,u,v} \right)^2 + \frac{1}{(h-1)w} \sum_{u,v} \mathbf{W}^{y}_{i,u,v} \left(\nabla_y \mathbf{P}_{i,u,v} \right)^2 \right]. \tag{17}$$

With $\gamma = 10.0$, strong image edges reduce smoothing across possible object or texture boundaries while promoting coherence elsewhere.

B.3 Fake-Image Sparsity Regularization

Because MIL only requires strong fake evidence in some patches, scores may increase broadly without additional constraints. We discourage this all-positive solution with

$$\mathcal{L}_{\text{sparse}} = \frac{1}{|\mathcal{F}|N} \sum_{i \in \mathcal{F}} \sum_{n=1}^{N} \sigma(s_{i,n}). \tag{18}$$

Its small weight regularizes the map without overriding MIL, allowing broad responses when supported by image-level evidence.

B.4 Patch-Level Contrastive Regularization

The MIL head outputs normalized patch embeddings $\mathbf{E} \in \mathbb{R}^{B \times D \times h \times w}$. From each image, we select the top ρ_{con} fraction by patch logit: selected fake-image patches are artifact-positive, and selected real-image patches are hard negatives. Let $\mathcal{A}$ denote this embedding set and $c_a \in \{0, 1\}$ its labels, with $c_a = 1$ for fake patches and $c_a = 0$ for real patches. For anchor a, the same-class positive set is

$$\mathcal{P}(a) = \{ p \in \mathcal{A} \,:\, p \neq a,\ c_p = c_a \}. \tag{19}$$

and the supervised contrastive loss is

$$\mathcal{L}_{\text{con}} = -\frac{1}{|\mathcal{V}|} \sum_{a \in \mathcal{V}} \frac{1}{|\mathcal{P}(a)|} \sum_{p \in \mathcal{P}(a)} \log \frac{\exp\left(\mathbf{e}_a^{\top} \mathbf{e}_p / \tau_{\text{con}} \right)}{\sum_{q \in \mathcal{A},\, q \neq a} \exp\left(\mathbf{e}_a^{\top} \mathbf{e}_q / \tau_{\text{con}} \right)}, \tag{20}$$

where $\mathcal{V}$ contains anchors with at least one positive sample. We use $\rho_{\text{con}} = 0.10$ and $\tau_{\text{con}} = 0.10$, and set the loss to zero when a minibatch has insufficient selected fake or real patches.

B.5 Optional Peak Separation Regularization

For a fake image i, sort patch logits as $s_{i,(1)} \geq s_{i,(2)} \geq \cdots \geq s_{i,(N)}$. The mean top response and mid-ranked background response are

$$A_i^{\text{top}} = \frac{1}{k_{\text{top}}} \sum_{n=1}^{k_{\text{top}}} s_{i,(n)}, \qquad k_{\text{top}} = \lceil \rho_{\text{top}} N \rceil. \tag{21}$$

$$A_i^{\text{rest}} = \frac{1}{k_{\max} - k_{\min}} \sum_{n=k_{\min}}^{k_{\max}} s_{i,(n)}, \tag{22}$$

where $k_{\min} = \lfloor \rho_{\min} N \rfloor$ and $k_{\max} = \lceil \rho_{\max} N \rceil$. The optional loss separates these responses by a margin:

$$\mathcal{L}_{\text{peak}} = \frac{1}{|\mathcal{F}|} \sum_{i \in \mathcal{F}} \max\left(0,\ m_{\text{peak}} - \left(A_i^{\text{top}} - A_i^{\text{rest}} \right) \right). \tag{23}$$

This term is implemented for ablation but disabled in the default configuration ($\lambda_{\text{peak}} = 0$).

B.6 Image-Wise Adaptive Calibration

Image-level supervision permits the absolute logit scale to vary across images. At inference only, we calibrate each prediction map using an adaptive raw-logit offset, then apply a fixed probability

threshold. Let $\mathbf{U}_i \in \mathbb{R}^{H\times W}$ be $\mathbf{S}_i$ bilinearly upsampled to the input resolution. Its uncalibrated probabilities and soft positive-response ratio are

$$\mathbf{P}_i^0 = \sigma(\mathbf{U}_i), \tag{24}$$

$$r_i = \frac{1}{HW}\sum_{u=1}^{H}\sum_{v=1}^{W}\mathbf{P}_{i,u,v}^0. \tag{25}$$

Here, r_i averages continuous probabilities rather than counting pixels above a binary threshold. The raw-logit statistics are

$$\mu_i = \frac{1}{HW}\sum_{u,v}\mathbf{U}_{i,u,v}, \qquad \sigma_i = \sqrt{\frac{1}{HW}\sum_{u,v}(\mathbf{U}_{i,u,v}-\mu_i)^2}. \tag{26}$$

Using a fixed reference response c, shared across images and domains, we compute

$$d_i = \mathrm{clip}\left(\frac{r_i - c}{\max(c,\epsilon)}, -1, 1\right), \tag{27}$$

$$k_i = \mathrm{clip}\left(k_0 - g d_i, k_{\min}, k_{\max}\right), \tag{28}$$

where k_0 is the base coefficient and g controls its adjustment. The adaptive offset and calibrated maps are

$$\tau_i = \mu_i + k_i\sigma_i. \tag{29}$$

$$\widetilde{\mathbf{U}}_i = s_{\mathrm{calib}}\frac{\mathbf{U}_i - \tau_i}{\sigma_i + \epsilon}, \qquad \widehat{\mathbf{M}}_i = \sigma\left(\widetilde{\mathbf{U}}_i\right). \tag{30}$$

We threshold $\widehat{\mathbf{M}}_i$ at $t = 0.5$. Thus, τ_i is an image-dependent centering offset in raw-logit space, whereas t is a global threshold in calibrated probability space; at $t = 0.5$, the raw-logit decision boundary is τ_i.

The default parameters are $k_0 = 0.9$, $s_{\mathrm{calib}} = 2.0$, $\epsilon = 10^{-6}$, $c = 0.08$, $g = 0.50$, $k_{\min} = 0.40$, and $k_{\max} = 1.20$. Calibration uses only the current image's prediction map, with no pixel-level ground truth, batch- or dataset-level statistics, threshold search, or target-domain tuning. The rule, parameters, and default threshold remain unchanged across in-domain, out-of-domain, and cross-dataset evaluations. Appendix D.5 varies only the final threshold to assess sensitivity.

C Implementation Details

ReGFLoW is implemented as a model module in IMDLBenCo, following the OpenSDI protocol and using the framework's data-loading, distributed-training, logging, and evaluation utilities.

Backbone configuration. The global encoder is a frozen CLIP ViT-L/14 [48]. The trainable local encoder follows the SideAdapterNetwork design in OpenSDI/IMDLBenCo and uses an ImageNet-pretrained MAE ViT-B/32 backbone [49]. Task-specific heads are trainable. Inputs are resized to 512×512 for the local branch and to the standard CLIP resolution for the global branch.

Training setup. Unless otherwise specified, training uses the OpenSDID-SD1.5 split with image-level real/fake labels; pixel-level masks are reserved for evaluation. We train ReGFLoW for 10 epochs on a single GPU with a batch size of 8. AdamW uses a learning rate of 5×10^{-5}, weight decay of 0.05, and no warmup; the random seed is 42. Augmentations include random scaling, flips, 90° rotations, mild brightness/contrast changes, image compression, and Gaussian blur.

Reconstruction prior. Each sample includes a single-channel 128×128 residual map for the reconstruction-guided modules. The map is resized as needed and converted to the input tensor type.

Optimization hyperparameters. Local-feature pooling gives a 16×16 MIL scoring grid. The scoring head has a hidden dimension of 256, adaptive MIL pooling uses sharpness $\kappa = 2.0$, and the ranking margin is 0.5. Loss definitions, weights, and hard-negative selection are specified in Appendix B. New heads use a learning-rate multiplier of 3.0.

Evaluation. Pixel-level F1 and IoU are computed from the calibrated localization map using the inference procedure in Appendix B.6. Image-level F1 and accuracy use the image classifier output.

D Additional Quantitative Analyses

We report pixel-level F1 (%) on OpenSDID. SD1.5 is the training domain; SD2.1, SDXL, SD3, and Flux.1 are unseen generators, whose unweighted mean is OOD Avg. The supervision comparisons use P+F, including partially edited and fully synthetic images during training and evaluation. The remaining analyses use P, evaluating partially edited images.

D.1 Effect of Supervision under Matched Backbones

To distinguish supervision-related effects from differences between independently designed architectures, we retain each backbone architecture and its non-head components while changing the localization head and supervision.

MaskCLIP backbone. MaskCLIP-WS replaces MaskCLIP's fully supervised segmentation head with our MIL head trained using image-level labels, retaining its backbone and non-head components [9].

ReGFLoW backbone. ReGFLoW-FS retains the ReGFLoW backbone and reconstruction guidance but replaces its MIL head with a segmentation head trained using pixel-level masks. All variants are trained on OpenSDID-SD1.5 under the same P+F protocol.

Table 8: Controlled supervision comparisons under P+F. Each pair shares the backbone architecture and non-head components but changes the localization head and supervision. Pixel-level F1 (%); bold marks the better result within each pair.

Supervision	Method	SD1.5	SD2.1	SDXL	SD3	Flux.1	OOD Avg.
(a) MaskCLIP backbone							
Full	MaskCLIP	**95.1**	**73.2**	20.8	17.7	4.8	**29.1**
Weak	MaskCLIP-WS	43.6	34.9	**28.9**	**29.4**	**20.7**	28.5
(b) ReGFLoW backbone with reconstruction guidance							
Full	ReGFLoW-FS	**92.22**	**65.26**	16.89	15.46	3.41	25.25
Weak	ReGFLoW	35.6	33.4	**30.1**	**32.2**	**29.4**	**31.3**

Table 8 shows the same pattern for both backbones: full supervision performs better on SD1.5 and SD2.1, whereas weak supervision improves SDXL, SD3, and Flux.1. ReGFLoW increases OOD Avg. from 25.25 to 31.3 relative to ReGFLoW-FS. MaskCLIP-WS has a comparable but slightly lower OOD Avg. than MaskCLIP (28.5 versus 29.1), reflecting the fully supervised model's SD2.1 advantage. These results support a contribution from the weakly supervised formulation beyond backbone choice or reconstruction guidance, but not a uniform gain across domains. Because the head and supervision change together, the comparisons assess the learning formulation rather than label granularity alone.

D.2 Reconstruction-Residual Localization Baseline

DIRE [6] and AEROBLADE [8] primarily address image-level detection; AEROBLADE also demonstrates qualitative localization of inpainted regions. To test our prior's standalone localization ability, we apply ReGFLoW's per-image normalization and final thresholding procedure ($t = 0.5$) directly to the SD1-VAE residual map, without training or dataset-specific tuning. This controlled baseline is not a reproduction of either complete method.

ReGFLoW exceeds residual thresholding by 15.6–30.8 F1 points across the five generators (Table 9). Under this inference rule, the residual is substantially weaker as a final localization map than as auxiliary guidance learned jointly with RGB evidence and artifact-centric MIL. The main-paper ablations examine the individual module contributions.

Table 9: Residual-only localization versus ReGFLoW under P, using the same per-image normalization and final thresholding procedure. Values are pixel-level F1 (%).

Method	SD1.5	SD2.1	SDXL	SD3	Flux.1
Recon. residual	18.5	22.1	17.4	18.5	13.6
ReGFLoW	**49.3**	**46.3**	**34.9**	**43.9**	**29.2**

D.3 Robustness to Alternative Reconstruction Priors

We replace only the inference-time prior with maps from DIRE-DDIM reconstruction [6] or an SDXL VAE [63]. The RGB input, SD1-VAE-trained checkpoint, learned parameters, and inference rule remain fixed, with no retraining, fine-tuning, adaptation, or prior-specific recalibration. This tests train–test mismatch in the auxiliary prior.

Table 10: Inference-time prior replacement under P, with the RGB input and learned parameters unchanged. Values are pixel-level F1 (%).

Reconstruction prior	SD1.5	SD2.1	SDXL	SD3	Flux.1
DIRE-DDIM	46.7	43.7	32.7	40.8	25.9
SDXL VAE	47.0	43.4	33.5	41.3	26.2
SD1 VAE (default)	**49.3**	**46.3**	**34.9**	**43.9**	**29.2**

The alternative priors reduce F1 by 1.4–3.3 points (Table 10). Although the default performs best, both replacements retain much of the localization performance, indicating tolerance to the tested alternatives rather than exclusive dependence on the training reconstructor.

D.4 Robustness to Resizing and Input Reconstruction

We extend the main-paper blur and JPEG evaluations to $0.5\times$ resizing and *edit-then-reconstruct*, which reconstructs the edited input with an SDXL VAE before inference. The checkpoint and inference parameters remain fixed, without retraining or transformation-specific tuning. Unlike prior replacement in Appendix D.3, edit-then-reconstruct changes the observed image itself.

Table 11: Input-transformation robustness under P. Edit-then-reconstruct applies an SDXL VAE to the edited image before inference. Values are pixel-level F1 (%).

Input condition	SD1.5	SD2.1	SDXL	SD3	Flux.1
Clean	49.34	46.27	34.86	43.90	29.18
Resize ($0.5\times$)	48.8	45.5	34.5	43.3	28.7
Edit-then-reconstruct	44.8	42.1	30.3	39.6	26.6

Resizing reduces F1 by less than 0.8 points per generator, while edit-then-reconstruct reduces it by approximately 2.6–4.6 points (Table 11). Together with the main-paper results, these tests cover compression, spatial smoothing, resizing, and reconstruction-based post-processing, complementing the separate auxiliary-prior mismatch analysis.

D.5 Sensitivity to the Final Decision Threshold

We retain the checkpoint and image-wise calibration in Appendix B.6 and vary only the final probability threshold t. This is a diagnostic sweep, not domain-specific threshold selection; $t = 0.5$ remains the shared default for all evaluations.

Performance is relatively stable over $t \in [0.4, 0.6]$, with at most 3.04 F1 points of within-generator variation (Table 12). Higher thresholds consistently reduce F1 as masks become more conservative. SDXL and Flux.1 favor $t = 0.4$, whereas SD1.5, SD2.1, and SD3 favor 0.5 or 0.6, confirming that the shared default does not maximize every domain's score.

Table 12: Final-threshold sensitivity under P, with the checkpoint and calibration fixed. Values are pixel-level F1 (%); bold marks the best result per generator, including ties. The shared default is $t = 0.5$.

Final threshold t	SD1.5	SD2.1	SDXL	SD3	Flux.1
0.4	48.20	45.68	**35.82**	43.42	**30.18**
0.5 (default)	**49.34**	**46.27**	34.86	**43.90**	29.18
0.6	**49.34**	45.63	32.78	43.17	27.26
0.7	47.75	43.33	29.21	40.65	24.06
0.8	43.40	38.01	23.07	34.97	18.86

E Additional Qualitative Results

E.1 Qualitative Analysis

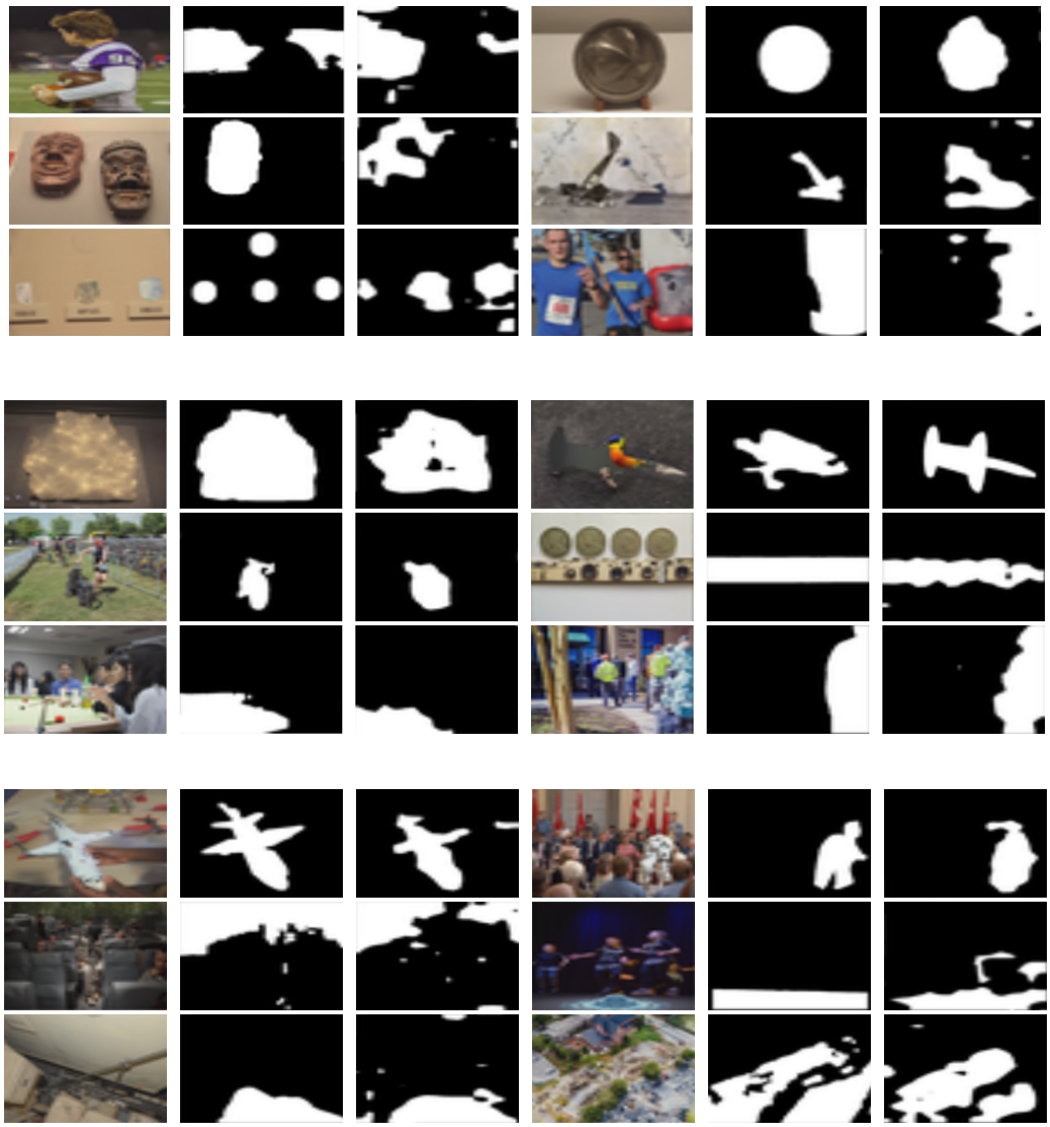

Figure 8: Additional out-of-domain localization results. Each triplet shows the input image, ground-truth mask, and ReGFLoW prediction from left to right.

Figure 8 extends the main-paper examples to mask-confined edits, boundary or out-of-mask traces, and non-semantic background changes. The predictions illustrate responses to broader artifact support rather than only object-aligned regions. Real-image false-positive controls and difficult cases involving tiny edits, compression artifacts, or high-frequency real textures illustrate the remaining limitations under cross-domain shifts.

F Limitations

ReGFLoW requires an additional reconstruction step to obtain its residual prior, increasing preprocessing or inference cost over RGB-only methods despite using a frozen autoencoder. Patch-level MIL can also miss edits smaller than or comparable to the patch resolution, or activate neighboring pixels, reducing localization precision and F1 for small, sharply defined masks. Finally, the controlled comparisons in Appendix D.1 reveal a source-domain/transfer trade-off: full supervision remains stronger on SD1.5 and SD2.1. The observed benefits therefore do not imply uniformly better performance on every domain.

G Related Work

G.1 From Image-Level AI-Generated Image Detection to Pixel-Level Localization

DIRE uses diffusion reconstruction error for image-level detection, based on the observation that generated images can be reconstructed more accurately than real images [6]. AEROBLADE uses autoencoder reconstruction error for training-free detection and demonstrates qualitative localization of inpainted regions [8]. RINE exploits intermediate CLIP representations rather than only final semantic features [7]. These methods motivate reconstruction behavior and pretrained features as forensic evidence, but an image-level authenticity score alone does not identify a small or spatially diffuse edited region. Dense localization requires spatial predictions; the residual-only comparison in Appendix D.2 tests our prior under this protocol.

G.2 From Conventional Image Manipulation Localization to Diffusion-Edited Localization

Conventional localization addresses splicing, copy-move, removal, and inpainting. IML-ViT emphasizes high-resolution processing, multi-scale features, and manipulation-edge supervision [18]. TruFor fuses RGB and learned noise-sensitive fingerprints with a transformer to predict localization, integrity, and reliability [19]. HiFi-IFDL learns hierarchical forgery attributes across manipulation types and generation sources for detection and localization [20].

Such methods often exploit boundaries, high-frequency inconsistencies, camera/noise patterns, or differences between authentic and manipulated regions. Diffusion editing can weaken these cues by blending or harmonizing nearby content, while object replacement, background generation, texture edits, and lighting changes need not have clear boundaries. TGIF evaluates this transition with text-guided inpainting forgeries [10]: conventional localizers can detect spliced regions but struggle with fully regenerated images, whereas synthetic-image detectors recognize regeneration without localizing the inpainted region.

G.3 Diffusion-Era Localization Benchmarks and the Need for Weak Supervision

X-Edit localizes text-guided edits using diffusion-inversion features and paired original/edited images with masks [11]. DEAL-300K combines instruction generation, mask-free editing, and an annotation pipeline to obtain pixel-level labels [12]. SIDA provides a multimodal detection, localization, and explanation framework with annotated social-media synthetic and tampered images [13]. TGIF2 extends inpainting evaluation to newer generators, including FLUX.1, and random non-semantic masks that probe object-centric biases [14]. BR-Gen targets underrepresented background regions such as sky, ground, wall, grass, and vegetation [15]. RADAR introduces BBC-PAIR, containing manipulations from 28 diffusion models, and combines semantic and geometric foundation-model features with contrastive learning [16]. OpenSDI studies open-world detection and localization through MaskCLIP, aligning CLIP with MAE for globally and locally manipulated images [9]. Despite these advances, reliance on paired images, dense masks, or large-scale annotation restricts training to sources where such supervision is available.

Weak supervision reduces this dependence. WSCL learns from image-level labels through multi-source and inter-patch consistency, using RGB, SRM, and Bayar streams for generic manipulations [21]. DOLOS compares explanation-, attention-, and local-score-based approaches for weakly supervised localization of diffusion-generated faces under controlled generator settings [22].

ReGFLoW learns from image-level labels for general diffusion-generated and diffusion-edited images. It combines dense reconstruction guidance with artifact-centric MIL instead of object-centric pseudo masks, manipulation boundaries, or camera/noise priors.

H Existing Assets and Licenses

Table 13 lists the datasets, codebases, pretrained models, and libraries used for research, with their original citations and available license terms.

Table 13: Existing assets used in this work.

Asset	**Usage in this work**	**License / terms**
OpenSDID / OpenSDI [9]	Training and evaluation dataset for diffusion-generated and diffusion-edited image detection/localization.	CC BY-SA 4.0; academic use.
IMDLBenCo [64]	Training, evaluation, logging, and metric computation framework.	CC BY 4.0.
CLIP [48]	Frozen global visual encoder.	MIT License.
MAE [49]	Initialization for the local artifact encoder.	CC BY-NC 4.0.
Stable Diffusion VAE / latent diffusion model [2]	Frozen reconstruction model used to compute the reconstruction residual prior.	CreativeML OpenRAIL-M.
Hugging Face Diffusers	Software library used for loading or running diffusion/VAE components.	Apache License 2.0.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes].

   Justification: The abstract, introduction, and Section 2 clearly state the paper's main claims: the limitations of mask-supervised diffusion-edited localization, the need for weak supervision, and the proposed task-specific reconstruction-guided MIL framework. These claims are supported by the method and experiments in Sections 3 and 4.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes].

   Justification: We discuss limitations in Appendix F, including the additional cost of computing reconstruction residuals and the difficulty of precisely localizing very small edits under patch-level weak supervision.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [N/A]

Justification: The paper does not present formal theoretical results, theorems, or proofs. The contribution is an empirical method evaluated through experiments.

Guidelines:

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

4. **Experimental result reproducibility**

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

Answer: [Yes].

Justification: The paper describes the model architecture, training objective, datasets, evaluation metrics, and inference procedure needed to reproduce the main results. Additional implementation details are provided in Appendix C.

Guidelines:

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

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [No].

   Justification: We do not provide open access to the full code at submission time. The paper provides implementation details for reproducibility, and we plan to release the code publicly upon publication.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes].

   Justification: Section 4.1 describes the datasets, supervision setting, baselines, and evaluation metrics. Appendix C and Appendix B provide additional training, loss, and inference details.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [No].

   Justification: We do not report error bars or confidence intervals because repeating all baseline and cross-domain evaluations over multiple seeds is computationally expensive. Instead, we use a fixed evaluation protocol and compare methods consistently across multiple in-domain and out-of-domain settings.

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes].

   Justification: Appendix C reports the distributed training setup, GPU configuration, and approximate training and evaluation time used for the main experiments.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes].

   Justification: We have reviewed the NeurIPS Code of Ethics and believe that the research conforms to it. The work uses existing datasets and models for forensic localization research and preserves anonymity in the submission.

   Guidelines:

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

10. **Broader impacts**

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

Answer: [No].

Justification: The paper does not include a separate broader impact discussion. The work is intended to support forensic analysis of AI-generated or edited images, but we do not provide a dedicated discussion of positive and negative societal impacts.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

Answer: [N/A].

Justification: The paper does not release a high-risk generative model, pretrained language model, or newly scraped dataset. The method is designed for forensic localization rather than content generation.

Guidelines:

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

12. **Licenses for existing assets**

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

Answer: [Yes].

Justification: We cite and credit the existing datasets, codebases, pretrained models, and external benchmarks used in this work. Appendix H summarizes their licenses and terms of use where available.

Guidelines:

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

13. **New assets**

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

Answer: [Yes].

Justification: The paper introduces the ReGFLoW implementation as a new research asset. We provide implementation details in Appendix C and plan to release the code publicly upon publication with documentation for training and evaluation.

Guidelines:

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

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

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

Answer: [N/A].

Justification: The paper does not involve crowdsourcing experiments or research with human subjects.

Guidelines:

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

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

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

    Answer: [N/A].

    Justification: The paper does not involve crowdsourcing experiments or research with human subjects, so IRB approval or equivalent review is not applicable.

    Guidelines:

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

16. **Declaration of LLM usage**

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

    Answer: [N/A].

    Justification: The core method, experiments, and scientific conclusions do not use LLMs as an important, original, or non-standard component. Any language assistance, if used, is limited to writing, editing, or formatting.

    Guidelines:

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