From Masks to Pixels and Meaning: A New Taxonomy, Benchmark, and Metrics for VLM Image Tampering

Xinyi Shang[1,2,*], Yi Tang[1,*], Jiacheng Cui[1,*], Ahmed Elhagry[1], Salwa K. Al Khatib[1], Sondos Mahmoud Bsharat[1], Jiacheng Liu[1], Xiaohan Zhao[1], Jing-Hao Xue[2], Hao Li[1], Salman Khan[1], and Zhiqiang Shen[1,†]

[1] Mohamed bin Zayed University of Artificial Intelligence, United Arab Emirates
[2] University College London, United Kingdom

Abstract. Existing tampering benchmarks largely rely on object masks, which severely misalign with the true edit signal: many pixels inside a mask are untouched or only trivially modified, while subtle yet consequential edits outside the mask are treated as natural. We reformulate VLM image tampering from coarse region labels to a pixel-grounded, meaning and language-aware task. First, we introduce a taxonomy spanning edit primitives (replace/remove/splice/inpaint/attribute/colorization, etc.) and their semantic class of tampered object, linking low-level changes to high-level understanding. Second, we release a new benchmark with per-pixel tamper maps and paired category supervision to evaluate detection and classification within a unified protocol. Third, we propose a training framework and evaluation metrics that quantify pixel-level correctness with localization to assess confidence or prediction on true edit intensity, and further measure tamper meaning understanding via semantics-aware classification and natural language descriptions for the predicted regions. We also re-evaluate the existing strong segmentation/localization baselines on recent strong tamper detectors and reveal substantial over- and under-scoring using mask-only metrics, and expose failure modes on micro-edits and off-mask changes. Our framework advances the field from masks to pixels, meanings and language descriptions, establishing a rigorous standard for tamper localization, semantic classification and description. Code and data are available at `https://github.com/VILA-Lab/PIXAR`.

Keywords: Pixel-wise Tampered Image Detection · Large-scale Image Tampering Benchmark

1 Introduction

> *What is real? How do you define 'real'?*
>
> Morpheus, *The Matrix (1999)*

* Equal contribution.
† Corresponding author. Email: Zhiqiang.Shen@mbzuai.ac.ae

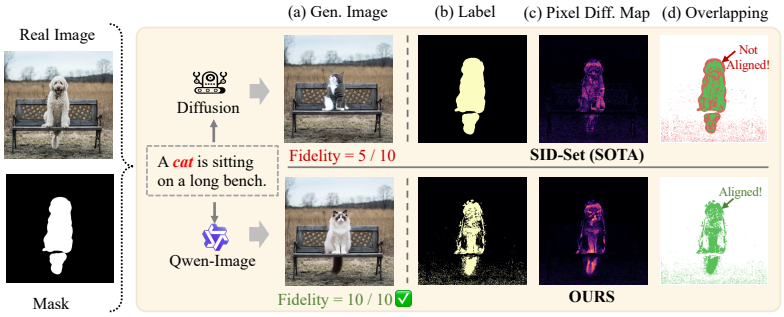

Fig. 1: Pitfalls of Current Benchmarks and Our Remedy. (a) They contain unrealistic samples. (b)–(d) Their widely adopted mask-based label contains large regions of pixels that are not aligned with the true generative pixels. In contrast, our pixel label is precisely aligned with the true generative pixels.

Advances in generative AI [5,32,33] have enabled the creation of photorealistic imagery, posing serious threats to digital media authenticity and trust [21, 25, 34]. Among these manipulations, fine-grained tampering [11, 12, 22] is particularly insidious, as it subtly modifies partial regions of real images while remaining imperceptible to both human and conventional forensic methods [10, 15]. Consequently, developing robust detectors for fine-grained tampering is both a critical research challenge and a societal necessity.

Yet most established benchmarks still annotate "where the edit is" using coarse object masks, as summarized in Tab. 1. This practice implicitly assumes that edits are spatially confined to a pre-specified region and that all pixels within that region are equally manipulated. In practice, however, the edit signal is neither spatially nor metrically uniform: many pixels inside a mask remain unchanged or only trivially perturbed, while visually consequential adjustments (e.g., relighting halos, color bleeding, deblurring, seam removal) frequently extend outside the mask. As a result, mask-only evaluation conflates unedited pixels with true tamper evidence and ignores off-mask artifacts, distorting both detector training and measurement.

We make this pixel-level real tamper explicit by contrasting per-pixel difference maps between original and tampered images in Fig. 1 (c), and visualize prior weaker solution SID-Set [15]'s mask-defined labels in Fig. 1 (b). Fig. 1 (d) reveals widespread misalignment (highlighted as red points) between the human-defined ground-truth and true pixel-level tamper, i.e., untouched pixels falsely labeled as "tampered" inside the mask and edited pixels incorrectly treated as "real" outside the mask. These label errors significantly penalize models for detecting genuine generative artifacts that fall beyond the mask boundary and, conversely, reward models that overfit to coarse shapes rather than the true edit footprint. Our analysis challenges the prevailing assumption that generative pipelines modify all masked pixels while preserving all unmasked areas, raising a fundamental question for the generative era: where, precisely, is the boundary between the "real" and the "generated", and how should benchmarks encode that boundary?

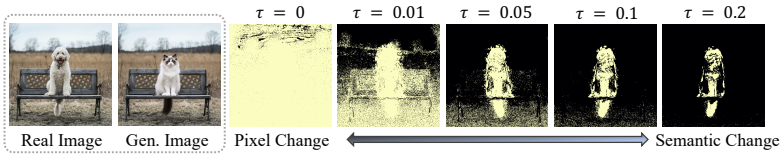

Fig. 2: Visualization of our pixel-level label under different τ.

To address this, we reformulate VLM image tampering as a **pix**el-grounded, meaning and text-**a**wa**r**e task and construct a new benchmark called `PIXAR`. Concretely, we derive a difference map between the original and edited images and convert it into a binary supervision signal via a tunable threshold τ. The resulting label map $\mathbf{M}_\tau$ captures the spatial support of the edit at a controllable intensity level (Fig. 2): small τ emphasizes sensitivity to micro-edits, while larger τ emphasizes conservative, high-confidence changes. This thresholded construction decouples where an edit occurs (localization) from how strongly it manifests (intensity), enabling principled sweeps over τ to select operating points that best correlate with human judgments and downstream scenario use cases. Consequently, our formulation aligns evaluation with the physical tampering signal rather than proxy geometries.

Building on this reformulation, we introduce a large-scale benchmark – over 380K carefully curated training image pairs with rich, standardized metadata, and a well-balanced test set containing 40K image pairs with their pixel-level and semantic-level labels. Each pair comprises a real source image, its tampered counterpart, the recommended binary pixel-level label map $\mathbf{M}_\tau$ from a default τ, and the raw per-pixel difference map from which alternative labels for other τ values can be derived. To ensure manipulation diversity, our pipeline integrates eight editing strategies instantiated via state-of-the-art both open- and closed-source generative models, including Flux.2 [16], Gemini 2.5 [9], Gemini 3 [8], GPT-image-1.5 [24], Qwen-Image [32], Seedream 4.5 [29]. The strategies span replace/remove/splice/inpaint/attribute/colorization and related primitives, and we manually annotate the semantic class of the tampered target to connect low-level footprints to high-level semantics. Finally, we design a rigorous multi-stage filtering pipeline to guarantee the fidelity of tampered images and the precision of the corresponding labels. These designs yield a dataset that is simultaneously pixel-faithful and semantically structured, supporting detection, localization, and semantic understanding within a single protocol.

In summary, our contributions are threefold:

1. We are the *first* to expose the core flaw of mask-based benchmarks, i.e. *misalignment with true generative edits*, and redefine tampering with pixel-level and semantic supervision to yield precise, faithful labels.
2. We build `PIXAR`, a large-scale, high-fidelity benchmark using state-of-the-art VLMs and human semantic label annotation, integrating 8 manipulation types with rigorous effectiveness and fidelity checks and accurate pixel annotations, establishing a principled foundation for the generative era.

Table 1: Details of recent publicly available tampering benchmarks. The **Task** column indicates the evaluation type, where "B-Cla." refers to binary classification ("real" vs. "tampered"), and "T-Loc." denotes tampering localization. "Multi-Object" means multiple objects within one image can be tampered with.

Dataset	Year	Task	Multi-Object	Fidelity Check	Ground Truth
ArtiFact [26]	2023	B-Cla.	✗	✗	-
TrainFors [22]	2024	B-Cla. & T-Loc.	✗	✗	Mask
SIDBench [28]	2024	B-Cla.	✗	✗	-
DiFF [4]	2024	B-Cla.	✗	✗	-
M3Dsynth [40]	2024	B-Cla. & T-Loc.	✗	✗	Mask
SemiTruths [25]	2024	B-Cla. & T-Loc.	✗	✗	Mask
AI-Face [18]	2025	B-Cla.	✗	✗	-
SID-Set [15]	2025	B-Cla. & T-Loc.	✗	✗	Mask
PIXAR	**2026**	B-Cla. & T-Loc.	✓	✓	Pixel & Semantics

3. We introduce a training framework and pixel-wise, realistic evaluation metrics, and extensively re-evaluate the state-of-the-art detectors on our PIXAR, revealing key limitations (e.g., micro- and off-mask failures) and setting stronger, more reliable baselines.

2 Related Work

Tampered Image Datasets. Early benchmarks primarily focus on full-image generation (e.g., text-to-image), training detectors for binary real-versus-fake classification [20, 38, 39]. However, as manipulations become subtler, recent research has shifted to fine-grained tampering detection [15]. Notably, SID-Set [15] uses Stable Diffusion–based inpainting [27] to construct a benchmark for tampering localization in social media images. Despite this progress, as summarized in Tab. 1, existing datasets largely use object masks as ground truth, leading to substantial misalignment with the true edit signal and fundamentally impairing detectors from learning true tampering footprints. In contrast, we redefine tampering with pixel-level and semantic supervision to yield precise, faithful labels.

Tampered Image Detection. Tampering detection is commonly formulated as a classification task using CNN- or Transformer-based architectures [3, 23]. For example, CNNSpot [31] learns universal CNN artifacts, and LGrad [30] maps images into gradient space to achieve model-driven generalization. More recently, Vision-Language Models have been introduced for tampering detection, such as SIDA [15], which fine-tunes LLaVA-based multimodal models, Antifake-Prompt [2], which employs prompt tuning to improve detection accuracy and cross-model generalization, and FakeShield [35], which leverages VLM and a decoupled architecture to provide explainable detection and precise localization across diverse forgery domains, supported by the multi-modal MMTD-Set.

3 Benchmark Construction

To rigorously train and evaluate image tampering detectors, we build PIXAR, a large-scale benchmark over 420K total image pairs with rich, standardized metadata. The construction of PIXAR is guided by three key principles: (i)

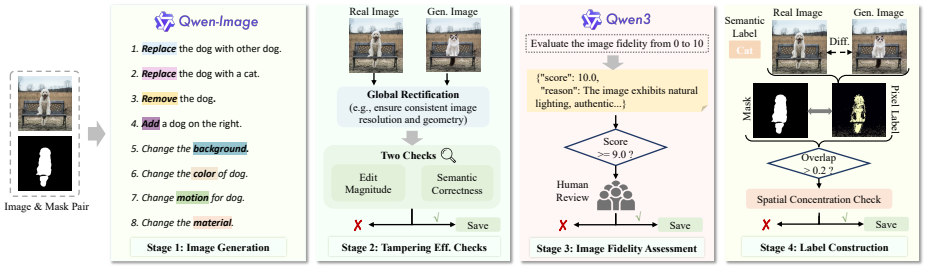

Fig. 3: Overview of PIXAR pipeline. It consists of four stages: image generation, tampering effectiveness checks, image fidelity assessment, and label construction.

Diversity: covering 8 representative tampering types that align well with real-world manipulation scenarios and demands; (ii) **Fidelity:** implementing rigorous fidelity checks to filter out low-fidelity samples; and (iii) **Precision:** ensuring precise labels. Accordingly, we propose a four-stage generation pipeline, as shown in Fig. 3. The following subsections describe each stage in detail, with more implementations provided in App. C.

3.1 Image Generation

Data Source and Generative Models. We use real source images from the COCO [19], a large-scale benchmark with diverse scenes, objects, and contexts. For training set, we employ Qwen-Image VLMs [32] due to their significant advances in complex text rendering and precise image editing. For test set, we use both open- and closed-source generative models, including Flux.2 [16], Gemini 2.5 [9], Gemini 3 [8], GPT-image-1.5 [24], Qwen-Image [32], Seedream 4.5 [29].

Diverse and Practical Tampering Types. Existing benchmarks mainly employ inter-class replacement for tampered image generation [15, 22, 40], which fails to reflect the complexity of real-world manipulations. To align well with real-world tampered scenarios and practical demands, we first analyzed large-scale Internet images to define 8 manipulation types. Qualitative examples are presented in Fig. 4. Detailed instructional design guidance, type distribution, and additional examples are provided in App. C.1.

Diverse Tampered Sizes and Complexities. To evaluate across difficulty levels, we control two factors: *tampered size* and *tampered complexity*. Tampered size measures the extent of pixel-level modification, where smaller edits leave subtler traces and are harder to detect, while larger edits yield more significant artifacts. Tampered complexity reflects the compositional structure of manipulations. Beyond the conventional single-object edits (Tab. 1), we apply a multi-object, sequential protocol to better reflect iterative real-world forgeries. Visualizations and size distributions are provided in Fig. 13b and Fig. 14 in the Appendix.

3.2 Tampering Effectiveness Checks

Generative models frequently exhibit failure modes, such as unintended global repainting or trivial perturbations. Therefore, we employ a rigorous filtering

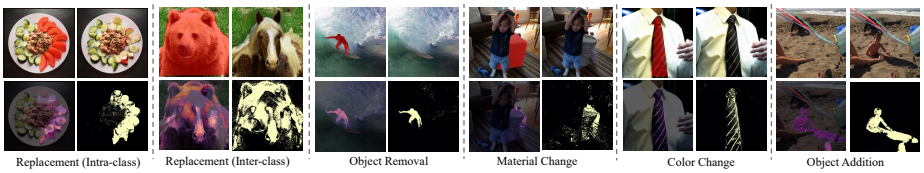

Fig. 4: Visualization of various tampering types in PIXAR. For each type, from top-left to bottom-right, the images show: original image with its mask, tampered image, pixel-difference map overlaid on the generated image, and our pixel-level label.

pipeline to remove ineffective tampered images. The pipeline consists of two sequential steps: (1) Global Rectification, which enforces a consistent pixel coordinate space between image pairs, and (2) Tampering Validity Checks, which verify whether the intended edit is both meaningful and semantically correct.

Global Rectification. In practice, many generative models (e.g., Gemini) do not support a fixed target resolution, which can lead to pixel-space misalignment between the generated image and the original image. Such misalignment makes pixel-level difference maps unreliable and can consequently corrupt the pixel label $\mathbf{M}_\tau$. To address this issue, we align I_{gen} to I_{orig} using feature matching, estimate a homography with RANSAC [7], and then recompute the difference map in the aligned coordinate space. The visualizations comparing the results before and after rectification are provided in Fig. 15 in Appendix.

Edit Magnitude and Semantic Correctness Checks. After rectification, we evaluate whether the model executed the intended edit on the original image. We observe three common failure cases: (i) *near-zero tampering*, where I_{gen} is almost identical to I_{orig}, yielding negligible signal in $\mathbf{M}_\tau$; (ii) *unintended global editing*, where large image regions are repainted beyond the target area, producing extensive noise in $\mathbf{M}_\tau$; and (iii) *unintended semantic edits*, where the visual change does not correspond to the text instruction. Accordingly, we assess validity from: (a) edit magnitude, to exclude trivial or overly global changes, and (b) semantic correctness, to ensure the visual manipulation matches the instruction. Failure cases and implementation details are provided in App. C.2.

3.3 Image Fidelity Assessment

The reliability of the benchmark critically depends on the fidelity of its samples. Low-quality samples provide limited information for model training and evaluation [13, 37]. To ensure high fidelity of images, we design a rigorous image fidelity assessment process that combines automated evaluation by the state-of-the-art VLM Qwen3 [36] with human expert evaluation.

Fig. 5: Comparison of fidelity between SID-Set [15] and our PIXAR.

Automated VLM Assessment. Each tampered image candidate is first evaluated by

Qwen3, which assigns a fidelity score from 0 to 10. Only tampered images achieving a score of ≥ 9 are shortlisted for subsequent human evaluation. This automated stage efficiently and effectively filters out low-fidelity images and thus improves the quality of our PIXAR.

Human Expert Review. Furthermore, we employ 10 human experts to manually review the generated images and remove those that appear unrealistic. Only samples that received a realism score of at least 4 out of 5 were retained. This human-in-the-loop evaluation ensures that subtle inconsistencies or semantic anomalies undetectable by automated models are effectively filtered out. We observed consistently high pass rates (90%) for intra-class replacement, splicing, inpainting, attribute modification, and colorization. In contrast, the inter-class replacement variant achieved a moderate pass rate of approximately 70%, while the removal type showed the lowest pass rate at around 55%. Representative examples of filtered and retained samples are provided in Fig. 16 in Appendix.

After the rigorous image filtering process, we randomly select a subset of tampered images from SID-Set [15] and from our PIXAR for comparison, as shown in Fig. 5. These samples from SID-Set generally appear visually unrealistic. In contrast, our dataset demonstrates significantly higher fidelity, establishing a more reliable and principled foundation for tampered image detection.

3.4 Label Construction

To remedy the flaw of the existing mask-based label, we redefine tampering with pixel-level and semantic supervision. Consequently, detectors trained on our PIXAR learn not only *where* image tampering occurs (local pixel detection) but also *what* the tampered content means (semantic understanding).

Pixel Label. Given a pair of real and tampered images (I_{orig}, I_{gen}), we compute a difference map $\mathbf{D}$ that quantifies the absolute pixel-wise discrepancy:

$$\mathbf{D}(\mathbf{x}, y) = |I_{orig}(\mathbf{x}, y) - I_{gen}(\mathbf{x}, y)|, \tag{1}$$

where $(\mathbf{x}, y)$ indexes pixel coordinates. We then obtain a binary supervision mask $\mathbf{M}_\tau$ by thresholding $\mathbf{D}$ with a tunable parameter τ:

$$\mathbf{M}_\tau(\mathbf{x}, y) = \mathbb{I}\big(\mathbf{D}(\mathbf{x}, y) > \tau\big), \tag{2}$$

where $\mathbb{I}(\cdot)$ denotes the indicator function. The resulting $\mathbf{M}_\tau$ captures the spatial support of the edit at a controllable intensity level: a small τ emphasizes sensitivity to micro-edits, while a larger τ emphasizes high-confidence modifications, as illustrated in Fig. 2. We provide more visualizations in App. A and a detailed analysis regarding the τ selection in Sec. 5.3.

Semantic Label. For each tampered object, we manually annotate its corresponding semantic class label (e.g., "cat" for the tampered image) and record this information as metadata associated with the image. Formally, let $\mathcal{C}$ be the set of object classes that may be tampered, with multi-label ground truth $\mathbf{y} \in \{0, 1\}^{|\mathcal{C}|}$, where $y_c = 1$ if class c contains a tampered instance.

Label Reliability Checks. Finally, we validate the reliability of the pixel-level label with respect to semantic supervision. In practice, low-quality pixel labels

arise from two issues: (i) *pixel–semantic inconsistency*, where the pixel change fails to reflect a valid semantic edit; and (ii) *poor spatial structure*, where the pixel label is semantically consistent yet overly dispersed (often dominated by background pixels), making it unreliable for supervision. We address these failures using two checks:

(1) Pixel–semantic consistency. As shown in Fig. 6, in some cases, replacing an object (e.g., the black oven in the first row) with a visually similar one may yield negligible L_1 differences due to near-identical color and texture. Consequently, the pixel-level label underestimates the true semantic change, causing a pixel–semantic mismatch. To ensure faithful supervision, we compute the overlap ratio between tampered pixels and the input mask, and discard samples with overlap < 0.2. More examples and a discussion of this threshold are provided in App. C.4.

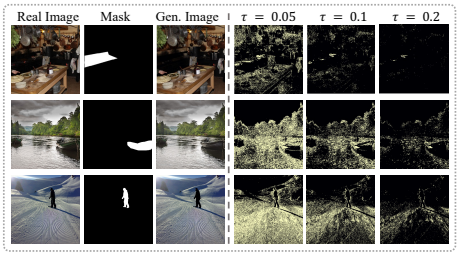

Fig. 6: Examples of pixel–semantic label inconsistency.

(2) Spatial concentration. Even when the semantic edit is correct, generation artifacts may introduce widespread background speckles, producing pixel labels that are scattered rather than object-shaped. Examples are visualized in Fig. 15 in Appendix. Such dispersed labels are semantically consistent but structurally uninformative. To remove them, we quantify the spatial concentration of $\mathbf{M}_\tau$ using: (i) a grid-based concentration ratio, defined as the fraction of grid cells required to cover 80% of tampered pixels; and (ii) a local density score, defined as the median density of tampered pixels within a small neighborhood. Based on these metrics, we discard *Diverse* samples and retain only *Concentrated* pixel labels. By filtering both inconsistent and dispersed cases, we obtain pixel supervision that is faithful to the edit and spatially well-formed.

3.5 Metadata

Each entry in our PIXAR benchmark encompasses four images accompanied by rich, standard metadata, providing comprehensive information about every stage of the tampering process, as illustrated in Fig. 7.

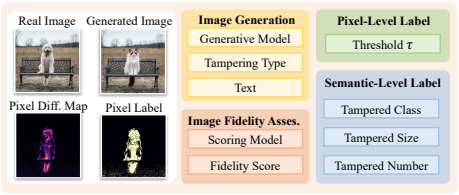

Fig. 7: Composition of PIXAR. Each entry includes four images with detailed metadata.

Specifically, each quadruple consists of: (i) a real source image, (ii) its tampered counterpart, (iii) the recommended binary pixel-level label map $\mathbf{M}_\tau$ from a default τ, and (iv) the raw per-pixel difference map from which alternative labels for other τ values can be derived. The accompanying metadata records detailed information on image generation, fidelity assessment process, and label construction. Notably, we provide text for manipulation description (details are provided in App. C.5). Moreover,

at the semantic level, beyond the semantic label, each tampered image is also accompanied by quantitative indicators such as the tampered size.

4 Training Framework

Our tamper detector f_θ produces (i) a per-pixel tamper logit map $\mathbf{S} \in \mathbb{R}^{H\times W}$ and probabilities $\widehat{\mathbf{M}} = \sigma(\mathbf{S})$, and (ii) a multi-label semantic logit vector $\mathbf{z} \in \mathbb{R}^{|\mathcal{C}|}$ with $\hat{\mathbf{y}} = \sigma(\mathbf{z})$. Here $\sigma(\cdot)$ is the element-wise sigmoid, also a global vector for real or tamper detection. (iii) a natural language description detailing the specific tampering artifacts. Fig. 8 illustrates our training framework, $\mathbf{h}$ represents the hidden feature vectors for different heads. We define the following losses:

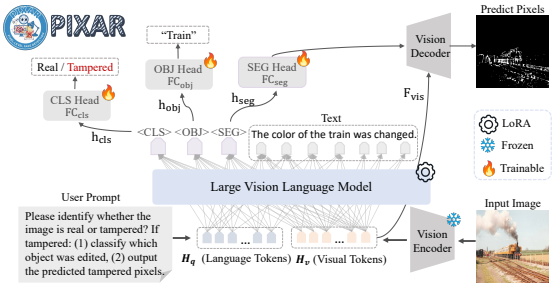

Fig. 8: Overview of training framework.

Multi-label semantic loss. Because one or multiple objects can be tampered in one image, we train with a sigmoid cross-entropy (per class):

$$\mathcal{L}_{\text{sem}} = -\frac{1}{|\mathcal{C}|}\sum_{c\in\mathcal{C}} \left[y_c \log \hat{y}_c + (1-y_c)\log\left(1-\hat{y}_c\right)\right], \tag{3}$$

where $y_c \in \{0,1\}$ is the ground-truth for class c, and $\hat{y}_c$ is the predicted probability for tampered area semantic label.

Pixel-wise BCE loss. We supervise the localization head with pixel-wise binary cross-entropy against our thresholded label $\mathbf{M}_\tau$:

$$\mathcal{L}_{\text{bce}} = -\frac{1}{HW}\sum_{i,j}[\mathbf{M}_\tau(i,j)\log\widehat{\mathbf{M}}_{ij} + (1-\mathbf{M}_\tau(i,j))\log(1-\widehat{\mathbf{M}}_{ij})], \tag{4}$$

where H, W are the height and width of an image.

Pixel-level DICE loss. Additionally, to further improve the localization of tampered pixels, we compute the DICE score [1] over connected-component masks. During training, we replace this surrogate mask with our pixel-level label $\mathbf{M}_\tau$ to more accurately reflect the true editing footprint:

$$\mathcal{L}_{\text{dice}} = 1 - \frac{2\sum_{i,j}\widehat{\mathbf{M}}_{ij}\mathbf{M}_\tau(i,j) + \varepsilon}{\sum_{i,j}\widehat{\mathbf{M}}_{ij} + \sum_{i,j}\mathbf{M}_\tau(i,j) + \varepsilon}, \tag{5}$$

with a small $\varepsilon > 0$ for numerical stability.

Global image-level detection loss. To decide whether an image is real or tampered[1], we use a global detection head driven by a special token $\langle\text{CLS}\rangle$. Let

[1] We do not consider the fully synthetic category used in SIDA as it is a special case when all pixels are tampered. This case is already covered by our pixel-level detection, which is distinct from mask-based training.

the backbone produce the last hidden states $\mathbf{H}^{\text{hid}} \in \mathbb{R}^{N\times d}$, we extract the ⟨CLS⟩ representation $h_{\text{cls}} = \mathbf{H}^{\text{hid}}[\text{CLS}] \in \mathbb{R}^d$, and feed it to a detection head F_{cls} to obtain global class logits and probabilities:

$$\begin{aligned} \mathbf{u} = F_{\text{cls}}\left(h_{\text{cls}}\right) \in \mathbb{R}^2, \quad \hat{\mathbf{p}} = \text{softmax}(\mathbf{u}), \\ \mathcal{L}_{\text{cls}} = \mathcal{L}_{\text{CE}}(\hat{\mathbf{p}}, \mathbf{d}), \end{aligned} \tag{6}$$

where $\mathcal{L}_{\text{cls}}$ is the global detection loss, and $\mathbf{d} \in \{0,1\}^2$ is the one-hot ground truth over {real, tampered}.

Tamper description generation loss. To provide a human-readable explanation of the manipulation, we generate a natural language description characterizing the tampered content (e.g., "A banana was added to the image."). We use a multimodal causal language model conditioned on the input image and the textual prompt to model the autoregressive likelihood of the target token sequence $T^* = (t_1^*, \ldots, t_L^*)$:

$$p_\phi(T^* \mid \mathbf{I}, P) = \prod_{i=1}^{L} p_\phi(t_i^* \mid t_{<i}^*, \mathbf{I}, P)\,, \mathcal{L}_{\text{text}} = -\sum_{i=1}^{L} \log p_\phi(t_i^* \mid t_{<i}^*, \mathbf{I}, P)\,, \tag{7}$$

where $\mathcal{L}_{\text{text}}$ is the standard language modeling loss and P is the input prompt.

Final objective. Our training objective is to minimize a weighted combination of the five losses:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{sem}}\mathcal{L}_{\text{sem}} + \lambda_{\text{bce}}\mathcal{L}_{\text{bce}} + \lambda_{\text{dice}}\mathcal{L}_{\text{dice}} + \lambda_{\text{text}}\mathcal{L}_{\text{text}} + \lambda_{\text{cls}}\mathcal{L}_{\text{cls}}. \tag{8}$$

where $\lambda_{\text{sem}}, \lambda_{\text{bce}}, \lambda_{\text{dice}}, \lambda_{\text{text}}, \lambda_{\text{cls}} > 0$ control the trade-offs between semantic understanding and pixel-accurate localization. We follow SIDA to choose values for λ_{bce} and λ_{cls}, and conduct ablations to determine the optimal value of λ_{sem}, λ_{dice}, and λ_{text} in Sec. 5.3. Unless stated otherwise, we use the recommended threshold $\tau = 0.05$ to form $\mathbf{M}_\tau$; sweeping τ at training or validation time provides a principled knob to balance micro-edit sensitivity against conservative high-precision localization. More analysis for τ selection is provided in Sec. 5.3.

5 Experiments

Extensive experiments are performed to demonstrate the efficacy of our training framework and the redefined ground truth. We also benchmark the performance of various state-of-the-art detectors on the challenging `PIXAR` test set to provide a rigorous comparative analysis.

5.1 Experimental Setup

Training and Test Dataset Details. For both the training and test datasets, we set the threshold to $\tau = 0.05$ by default, with results of other τ discussed in Tab. 6. To ensure a fair evaluation of detector performance, the test dataset is carefully constructed to maintain a balanced distribution across tampered classes, tampered types, and tampered sizes, resulting in 40K images with their pixel-level and semantic-level labels. The detailed procedure for constructing the test set is provided in App. D.

Table 2: Pixel-level Tampered Region Localization Results and Associated Semantic Prediction Accuracy. PIXAR-Lite models are fine-tuned on a subset of our training data that contains masks to guide the generation of tampered images. In this setting, LISA and SIDA utilize these masks as ground-truth, whereas our model is supervised by the pixel-difference map with a threshold of $\tau = 0.05$.

Methods	Semantic Classification		Pixel Localization				
	Top-1 Acc	Top-5 Acc	Recall	F1-Score	AUC	g-IoU	IoU
LISA-7B [17]	27.1	71.6	10.0	15.4	55.0	7.7	8.3
SIDA-7B [15]	27.1	71.9	15.0	21.1	55.0	10.7	11.8
PIXAR-7B-Lite (Ours)	28.2	75.0	26.4	26.1	55.2	14.3	15.0
PIXAR-7B (Ours)	**36.2**	**77.0**	**29.8**	**30.6**	**62.2**	**16.1**	**18.1**
LISA-13B [17]	30.6	75.1	11.4	17.3	55.6	9.0	9.5
SIDA-13B [15]	30.8	75.4	13.2	19.5	55.6	10.7	10.8
PIXAR-13B-Lite (Ours)	30.9	76.0	26.2	26.7	55.7	15.0	15.4
PIXAR-13B (Ours)	**37.4**	**76.0**	**33.6**	**32.3**	**62.2**	**17.8**	**19.3**

Implementation Details. Leveraging SIDA [15] and LISA [17], our model is fine-tuned on PIXAR to transition from coarse to precise pixel-level localization. Through joint supervision via spatial, multi-label semantic (Eq. 3), and text generation (Eq. 7) losses, the framework simultaneously produces refined tamper masks, categorizes affected objects, and generates descriptive explanations.

Pixel-level and Semantic Metrics. To evaluate our framework and redefined ground truth, we introduce a novel paradigm integrating pixel-level localization with semantic classification. Moving beyond conventional binary or coarse region-level metrics, this approach enables a holistic assessment of fine-grained precision and the interpretive understanding of tampered artifacts. We assess pixel-level localization using Recall, F1-Score, and AUC to evaluate the precision of delineated tampered regions. To rigorously measure spatial overlap, we employ image-level mean IoU (g-IoU) for per-sample robustness and dataset-level IoU (IoU) for overall pixel-wise accuracy. Furthermore, semantic alignment is quantified via Top-1 and Top-5 Accuracy, evaluating the model's proficiency in categorizing manipulated objects. By integrating spatial precision with semantic alignment, this interpretable framework provides a robust and holistic benchmark for fine-grained image forensics.

5.2 Evaluation on PIXAR

Semantic Alignment and Localization Results. Tab. 2 reports the detection performance on the PIXAR test set. We select LISA and the SOTA method SIDA [15] as baselines for comparison. The results show that replacing coarse masks with our pixel-level labels simultaneously improves the model's ability to accurately localize tampered regions and its semantic consistency in predicting the manipulation context. Notably, by merely refining the supervision signal, PIXAR-7B-Lite significantly outperforms SIDA-7B, elevating IoU from

Table 3: Performance comparison between PIXAR and FakeShield.

Methods	Pixel Localization				
	Recall	F1-Score	AUC	g-IoU	IoU
FakeShield [35]	10.7	17.0	52.0	8.4	9.3
PIXAR-7B	29.8	30.6	62.2	16.1	18.1
PIXAR-13B	**33.6**	**32.3**	**62.2**	**17.8**	**19.3**

Table 4: Comparison of existing deepfake detection methods and our models evaluated on the PIXAR test set.

Detector	Backbone	Data Distribution		Precision			Recall			F1-Score		
		GANs	Diffusion	Real	Fake	Overall	Real	Fake	Overall	Real	Fake	Overall
CnnSpot [31]	ResNet-50 [14]	✓	✗	15.2	84.4	49.8	99.4	0.6	50.0	26.4	1.2	13.8
AntifakePrompt [2]	InstructBLIP [6]	✓	✓	4.6	84.8	44.7	0.01	99.9	50.0	0.03	91.7	45.9
SIDA-7B [15]	LISA-7B [17]	✗	✓	15.8	86.7	51.3	88.2	13.6	50.9	26.8	23.4	25.1
PIXAR-7B	SIDA-7B [15]	✗	✓	**39.1**	**97.6**	**68.4**	**89.9**	**74.9**	**82.4**	**54.5**	**84.8**	**69.7**

Table 5: Comparison of detection performance across different generative sources.

Methods	Semantic Classification		Pixel Localization				
	Top-1 Acc	Top-5 Acc	Recall	F1-Score	AUC	g-IoU	IoU
GPT-Image-1.5 [24]	26.0	70.5	16.5	20.9	52.3	11.3	11.7
Seedream 4.5 [29]	29.9	73.1	27.8	26.8	57.8	13.9	15.5
Gemini 3 [8]	33.8	75.7	34.5	26.7	67.0	14.6	15.4
Gemini 2.5 [9]	34.0	76.9	23.7	29.5	70.2	16.1	17.3
Flux.2 [16]	37.1	79.7	25.7	31.3	56.3	16.7	18.6
Qwen-Image [32]	52.0	84.0	56.5	41.6	76.6	22.6	26.3

11.8% to 15.0% and Top-1 Acc from 27.1% to 28.2%. This performance is further bolstered in PIXAR-7B, which achieves an additional 8.0% gain when scaled to the full training set. Tab. 3 further benchmarks our approach against FakeShield [35], where PIXAR demonstrates a commanding lead across all evaluated metrics. Most notably, PIXAR-7B achieves a near-doubling of localization accuracy, elevating the IoU from 9.3% to 18.1%. Collectively, these findings corroborate the effectiveness of accurate pixel-level supervision in achieving high-precision localization and the robustness of our framework across various evaluation settings. We further provide a time cost comparison in App. G, demonstrating that our method maintains a runtime comparable to existing approaches, since the additional loss terms introduce negligible training overhead.

Binary Classification Performance. We evaluate a range of open-source deepfake detectors, CnnSpot [31], AntifakePrompt [2], and SIDA-7B [15] against our PIXAR models. As shown in Tab. 4, our models consistently outperform all baselines, highlighting their superior generalization, precise pixel-level localization, and robust binary classification.

Evaluation across Generative Models. To comprehensively evaluate the performance of PIXAR across various generative paradigms, we report the individual results for images generated by Qwen-Image [32], GPT-Image-1.5 [24], Gemini 2.5 [9], Gemini 3 [8], Seedream 4.5 [29], and Flux.2 [16] in Tab. 5. Notably, GPT emerges as the most challenging case, yielding only 11.7% IoU and 26.0% Top-1 accuracy. Conversely, images generated by Qwen prove significantly more susceptible to detection. We hypothesize that this performance discrepancy is rooted in a severe domain shift. Because the underlying training dataset is predominantly generated by Qwen, the detector tends to align its feature representations with Qwen-specific generation patterns.

5.3 Ablation Study

Influence of Different τ. To investigate how the training threshold τ influences tampered-pixel localization, we conduct an ablation study as shown in Tab. 6. The results indicate that increasing τ leads to a degradation in localization performance. This phenomenon can be attributed to the fact that a higher threshold progressively suppresses fine-grained pixel information, preserving primarily high-level semantic cues that resemble the masks used to generate images, as illustrated in Fig. 2. Consequently, the supervision signal becomes less informative and less reliable for guiding precise pixel localization. More analyses are provided in App. A.

Table 6: Pixel localization of models trained with different threshold τ, evaluated at a fixed threshold of $\tau = 0.05$.

	Semantic Classification		Pixel Localization				
	Top-1 Acc	Top-5 Acc	Recall	F1-Score	AUC	g-IoU	IoU
$\tau = 0.05$	**36.2**	**77.0**	**29.8**	**30.6**	**62.2**	**16.1**	**18.1**
$\tau = 0.1$	35.2	76.1	15.4	22.4	60.9	13.1	12.6
$\tau = 0.2$	34.2	75.9	9.6	16.0	61.2	9.5	8.7

Influence of λ_{sem}. Tab. 7 explores the impact of semantic loss weight λ_{sem}. We observe that both lower (0.1) and higher (1.0) values of λ_{sem} lead to a marginal decline in Top-1 accuracy. This suggests that $\lambda_{sem} = 0.5$ strikes the optimal balance, providing sufficient semantic supervision without overshadowing other task objectives. We therefore adopt $\lambda_{sem} = 0.5$ to ensure maximum multi-task synergy.

Table 7: Impact of Semantic Loss Weight.

	Semantic Classification		Pixel Localization				
	Top-1 Acc	Top-5 Acc	Recall	F1-Score	AUC	g-IoU	IoU
$\lambda_{sem} = 0.1$	35.1	76.2	29.7	30.3	62.2	15.9	17.9
$\lambda_{sem} = 0.5$	**36.2**	**77.0**	**29.8**	**30.6**	**62.2**	**16.1**	**18.1**
$\lambda_{sem} = 1.0$	35.2	76.2	29.7	30.5	62.0	16.0	18.0

Influence of λ_{text}. Tab. 8 investigates the trade-off controlled by λ_{text}. While a lower λ_{text} leads to suboptimal text generation, an excessively high value (e.g., $\lambda_{text} = 4.0$) tends to degrade semantic accuracy. We observe that $\lambda_{text} = 3.0$ strikes an optimal balance, delivering superior text quality without compromising core detection performance; thus, it is adopted as our default configuration.

Table 8: Impact of Text Loss Weight.

	Semantic Classification		Pixel Localization		Text Quality
	Top-1 Acc	Top-5 Acc	g-IoU	IoU	Css
$\lambda_{text} = 2.0$	35.6	76.7	16.1	18.1	0.51
$\lambda_{text} = 3.0$	**36.2**	**77.0**	**16.1**	**18.1**	**0.75**
$\lambda_{text} = 4.0$	35.9	76.9	16.1	18.1	0.75

Influence of λ_{dice}. As summarized in Tab. 9, we conduct an ablation study to evaluate the contribution of Dice loss. The results demonstrate that incorporating λ_{dice} simultaneously enhances localization precision and semantic classification accuracy. This indicates that Dice loss provides superior spatial supervision, which refines mask boundaries and strengthens discriminative feature extraction.

Table 9: Impact of Dice Loss Weight λ_{dice}.

	Semantic Classification		Pixel Localization				
	Top-1 Acc	Top-5 Acc	Recall	F1-Score	AUC	g-IoU	IoU
$\lambda_{dice} = 0.0$	35.3	76.3	12.9	19.5	61.6	7.4	10.8
$\lambda_{dice} = 0.5$	36.0	76.6	22.8	27.3	62.0	13.5	15.8
$\lambda_{dice} = 1.0$	**36.2**	**77.0**	**29.8**	**30.6**	**62.2**	**16.1**	**18.1**

5.4 Analysis

Visualization. To visually compare the prediction results between `PIXAR` and SIDA (after fine-tuning on our dataset), we provide a visualization in Fig. 9. Within these red dashed boxes, we illustrate both true positives and false negatives. For true positives, the closer the predicted tampered pixels are to the actual pixel

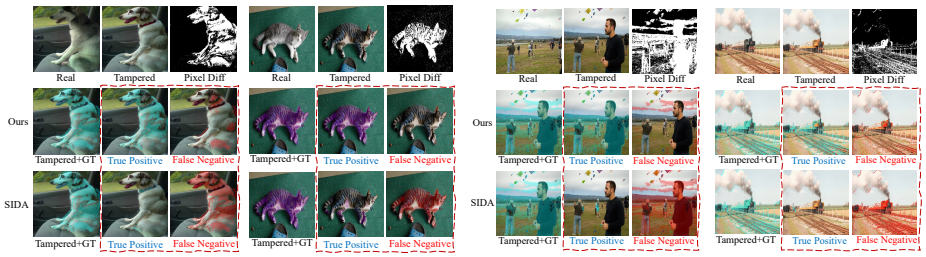

Fig. 9: Visualization comparison of prediction results between PIXAR and SIDA [15]. The red dashed boxes show different prediction focuses compared to SIDA.

differences, the better. False negatives refer to tampered pixels that the model fails to detect - the fewer, the better. The figure clearly demonstrates that *using masks as supervision* ***fails*** *to effectively recover the actual tampered regions*, most of the manipulated areas are missed (see the false negatives of SIDA), and only a small portion of the tampered regions are correctly detected. In contrast, our model exhibits a strong ability to accurately localize the tampered regions, with only a limited number of false negatives, further validating the effectiveness of the pixel-difference map over the ambiguous mask-based supervision in localization.

User study. To complement our quantitative metrics, we conduct a user study to evaluate the perceptual quality of tampered images in PIXAR. A random subset of our dataset is selected for this study, consisting of 1,000 images in total, including 500 real and 500 tampered samples. 10 participants are asked to perform two tasks: (1) classify whether the image is real or tampered, and (2) if tampered, localize the manipulated regions by drawing bounding boxes around the perceived alterations. As shown in Tab. 10, participants exhibit *low performance* in both binary classification and fine-grained localization, indicating that the tampered images in our dataset achieve high visual realism.

Table 10: Results of user study.

	Binary Classification			**Localization**		
	Precision	Recall	F1-Score	Recall	F1-Score	IOU
Human	22.2	55.5	31.0	17.4	18.8	10.7

6 Conclusion

In this work, we revisited VLM tampering as a pixel-grounded, meaning and language-aware task by deriving per-pixel difference maps and thresholding with τ to obtain controllable labels $\mathbf{M}_\tau$. We also release PIXAR, a high-fidelity, large-scale benchmark of more than 420K image pairs built with 8 diverse manipulations, providing original and tampered images, rich metadata, raw difference maps, recommended $\mathbf{M}_\tau$, and language descriptions for flexible supervision. We further introduced a pixel-aware training framework for localization with semantics-aware classification and natural language descriptions, and show that state-of-the-art detectors are ill-scored under mask-only protocols, especially on micro-edits and off-mask changes, establishing a more realistic and reliable standard for fine-grained tamper detection and understanding.

Acknowledgments

This work is supported by the United Al Saqer Group Grant.

References

1. Azad, R., Heidary, M., Yilmaz, K., Hüttemann, M., Karimijafarbigloo, S., Wu, Y., Schmeink, A., Merhof, D.: Loss functions in the era of semantic segmentation: A survey and outlook. arXiv preprint arXiv:2312.05391 (2023)
2. Chang, Y.M., Yeh, C., Chiu, W.C., Yu, N.: Antifakeprompt: Prompt-tuned vision-language models are fake image detectors. arXiv preprint arXiv:2310.17419 (2023)
3. Chen, L., Zhang, Y., Song, Y., Liu, L., Wang, J.: Self-supervised learning of adversarial example: Towards good generalizations for deepfake detection. In: CVPR. pp. 18710–18719 (2022)
4. Cheng, H., Guo, Y., Wang, T., Nie, L., Kankanhalli, M.: Diffusion facial forgery detection. In: ACM MM. pp. 5939–5948 (2024)
5. Comanici, G., Bieber, E., Schaekermann, M., Pasupat, I., Sachdeva, N., Dhillon, I., Blistein, M., Ram, O., Zhang, D., Rosen, E., et al.: Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. arXiv preprint arXiv:2507.06261 (2025)
6. Dai, W., Li, J., Li, D., Tiong, A., Zhao, J., Wang, W., Li, B., Fung, P.N., Hoi, S.: Instructblip: Towards general-purpose vision-language models with instruction tuning. NeurIPS **36**, 49250–49267 (2023)
7. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM **24**(6), 381–395 (1981)
8. Google AI for Developers: Gemini 3 pro image preview (gemini api model documentation). `https://ai.google.dev/gemini-api/docs/models/gemini-3-pro-image-preview` (2026), model ID: gemini-3-pro-image-preview. Accessed: 2026-02-27
9. Google Cloud: Gemini 2.5 flash image (vertex ai model documentation). `https://docs.cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-5-flash-image` (2026), model ID: gemini-2.5-flash-image. Accessed: 2026-02-27
10. Guo, X., Liu, X., Masi, I., Liu, X.: Language-guided hierarchical fine-grained image forgery detection and localization. International Journal of Computer Vision **133**(5), 2670–2691 (2025)
11. Guo, X., Liu, X., Ren, Z., Grosz, S., Masi, I., Liu, X.: Hierarchical fine-grained image forgery detection and localization. In: CVPR. pp. 3155–3165 (2023)
12. Gupta, P., Ghosh, S., Gedeon, T., Do, T.T., Dhall, A.: Multiverse through deepfakes: The multifakeverse dataset of person-centric visual and conceptual manipulations. arXiv preprint arXiv:2506.00868 (2025)
13. Gupta, V., Ross, C., Pantoja, D., Passonneau, R.J., Ung, M., Williams, A.: Improving model evaluation using smart filtering of benchmark datasets. In: NAACL. pp. 4595–4615 (2025)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
15. Huang, Z., Hu, J., Li, X., He, Y., Zhao, X., Peng, B., Wu, B., Huang, X., Cheng, G.: Sida: Social media image deepfake detection, localization and explanation with large multimodal model. In: CVPR. pp. 28831–28841 (2025)

16. Labs, B.F.: FLUX.2: Frontier Visual Intelligence. https://bfl.ai/blog/flux-2 (2025), accessed 2026-02-27
17. Lai, X., Tian, Z., Chen, Y., Li, Y., Yuan, Y., Liu, S., Jia, J.: Lisa: Reasoning segmentation via large language model. In: CVPR. pp. 9579–9589 (2024)
18. Lin, L., Santosh, S., Wu, M., Wang, X., Hu, S.: Ai-face: A million-scale demographically annotated ai-generated face dataset and fairness benchmark. In: CVPR. pp. 3503–3515 (2025)
19. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV. pp. 740–755 (2014)
20. Lu, Z., Huang, D., Bai, L., Qu, J., Wu, C., Liu, X., Ouyang, W.: Seeing is not always believing: Benchmarking human and model perception of ai-generated images. vol. 36, pp. 25435–25447 (2023)
21. Monteith, S., Glenn, T., Geddes, J.R., Whybrow, P.C., Achtyes, E., Bauer, M.: Artificial intelligence and increasing misinformation. The British Journal of Psychiatry **224**(2), 33–35 (2024)
22. Nandi, S., Natarajan, P., Abd-Almageed, W.: Trainfors: A large benchmark training dataset for image manipulation detection and localization. In: ICCV. pp. 403–414 (2023)
23. Nguyen, H.H., Yamagishi, J., Echizen, I.: Capsule-forensics: Using capsule networks to detect forged images and videos. In: ICASSP. pp. 2307–2311 (2019)
24. OpenAI: GPT Image 1.5 (model documentation). https://developers.openai.com/api/docs/models/gpt-image-1.5 (2026), model ID: gpt-image-1.5. Accessed: 2026-02-24
25. Pal, A., Kruk, J., Phute, M., Bhattaram, M., Yang, D., Chau, D.H., Hoffman, J.: Semi-truths: A large-scale dataset of ai-augmented images for evaluating robustness of ai-generated image detectors. NeurIPS **37**, 118025–118051 (2024)
26. Rahman, M.A., Paul, B., Sarker, N.H., Hakim, Z.I.A., Fattah, S.A.: Artifact: A large-scale dataset with artificial and factual images for generalizable and robust synthetic image detection. In: ICIP. pp. 2200–2204 (2023)
27. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR. pp. 10684–10695 (2022)
28. Schinas, M., Papadopoulos, S.: Sidbench: A python framework for reliably assessing synthetic image detection methods. In: Proceedings of the 3rd ACM International Workshop on Multimedia AI against Disinformation. pp. 55–64 (2024)
29. Seedream, T., Chen, Y., Gao, Y., et al.: Seedream 4.0: Toward next-generation multimodal image generation. arXiv preprint arXiv:2509.20427 (2025)
30. Tan, C., Zhao, Y., Wei, S., Gu, G., Wei, Y.: Learning on gradients: Generalized artifacts representation for gan-generated images detection. In: CVPR. pp. 12105–12114 (2023)
31. Wang, S.Y., Wang, O., Zhang, R., Owens, A., Efros, A.A.: Cnn-generated images are surprisingly easy to spot... for now. In: CVPR. pp. 8695–8704 (2020)
32. Wu, C., Li, J., Zhou, J., Lin, J., Gao, K., Yan, K., Yin, S.m., Bai, S., Xu, X., Chen, Y., et al.: Qwen-image technical report. arXiv preprint arXiv:2508.02324 (2025)
33. Xia, Z., Han, D., Han, Y., Pan, X., Song, S., Huang, G.: Gsva: Generalized segmentation via multimodal large language models. In: CVPR. pp. 3858–3869 (2024)
34. Xu, D., Fan, S., Kankanhalli, M.: Combating misinformation in the era of generative ai models. In: ACM MM. pp. 9291–9298 (2023)
35. Xu, Z., Zhang, X., Li, R., Tang, Z., Huang, Q., Zhang, J.: Fakeshield: Explainable image forgery detection and localization via multi-modal large language models. arXiv preprint arXiv:2410.02761 (2024)

36. Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al.: Qwen3 technical report. arXiv preprint arXiv:2505.09388 (2025)
37. Zhang, C., Pan, F., Kim, J., Kweon, I.S., Mao, C.: Imagenet-d: Benchmarking neural network robustness on diffusion synthetic object. In: CVPR. pp. 21752–21762 (June 2024)
38. Zhong, N., Xu, Y., Qian, Z., Zhang, X.: Rich and poor texture contrast: A simple yet effective approach for ai-generated image detection. CoRR (2023)
39. Zhu, M., Chen, H., Yan, Q., Huang, X., Lin, G., Li, W., Tu, Z., Hu, H., Hu, J., Wang, Y.: Genimage: A million-scale benchmark for detecting ai-generated image. vol. 36, pp. 77771–77782 (2023)
40. Zingarini, G., Cozzolino, D., Corvi, R., Poggi, G., Verdoliva, L.: M3dsynth: A dataset of medical 3d images with ai-generated local manipulations. In: ICASSP. pp. 13176–13180 (2024)