FORGEPAN-5.9M: A GENERALIZABLE BENCHMARK FOR IMAGE FORGERY DETECTION AND LOCALIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

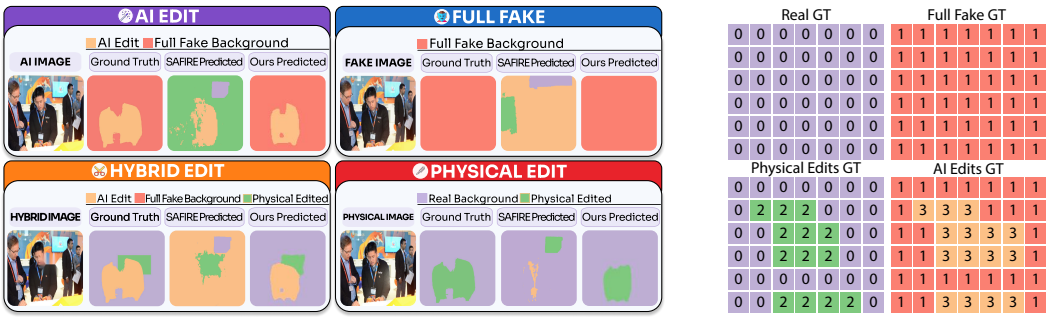
Rapid advancements in AI-generated content (AIGC) are blurring the boundaries of reality and the synthetic. AIGC can be a fully generated image, partially manipulated by either AI or physical editing tools, or a mixture of both, which we term "Hybrid Edit". Despite the increasing volume of AIGC detection benchmarks, prior efforts struggle to generalize to the granularity of manipulation types, such as detectors being reliable for fully generated images may perform poorly on partially edited images, or vice versa, nor is there a unified detection model that can generalize well across different real-world manipulations in terms of both detection and localization. Existing methods generalize poorly to hybrid edits, which are multiturn edited images with both physical edits and AI edits. We first introduce ForgePan-5.9M, a large-scale detection and localization benchmark featuring data-aligned samples across multiple manipulation types, such as AI editing, physical copy/move, and full-image generation, enabling unified evaluation of both manipulation detection and localization. This aligned dataset aided with proposed ForgeFormer that learns pixel-level intrinsic fingerprints associated with diffusion, GANs, or physically edited forgeries and ForgeFormer simultaneously achieves state-of-the-art in identifying and localizing unseen fully generated AI images and partial manipulations by either AI or physical edits and generalizes well on hybrid edits. Notably, ForgeFormer achieves 1st rank with 87% balanced accuracy on the IEDAL2 challenge dataset, further validating its effectiveness in realistic, heterogeneous forgery scenarios.

1 INTRODUCTION

In recent years, image manipulation has become increasingly accessible due to the proliferation of tools for generating and editing fake images. An image circulating on social media may be entirely AI-generated, subtly altered using a diffusion Rombach et al. (2022a); Saharia et al. (2022); Nichol et al. (2022) or GAN-based inpainting model Suvorov et al. (2022) to distort its contextual meaning, minimally retouched with professional image-editing software, or a combination of inpainting and physical editing. Existing forensics tools for edited image detection and localization, such as TruFor Guillaro et al. (2023), MVSS-Net Dong et al. (2023), PSCC-Net Liu et al. (2022b), CAT-Net Kwon et al. (2021), SIDA Huang et al. (2025), and RADAR Costanzino et al. (2026), accept the localization of fake regions as a binary segmentation problem and collapse when subjected to hybrid edits. Subsequently, methods such as SAFIRE Kwon et al. (2025) and DiffSeg30k Ci et al. (2025) have been formulated to perform instance-level segmentation for localizing multiple forgeries generated by diffusion-based approaches such as SDXL Podell et al. (2024) or SDV2.1 Rombach et al. (2022b), which may collapse when the background is real or fake, or the foreground is edited with multiple forgeries, such as physical edits or AI edits. This binary abstraction is no longer a sufficient representation of the threat model: the *nature* of the manipulation, whether fully synthetic, diffusion-inpainted, GAN edited, physically retouched, or some combination thereof, and the coexistence of *multiple* manipulation types within a single image are critical for forensic attribution, chain of custody analysis, and downstream content moderation decisions. Further related work is given in Appendix. C.

A common challenge in detecting and localizing fake images is generalization, particularly to out-of-distribution (OOD) samples, because forensic cues are often imperceptible to human observers Guillaro et al. (2025). Alignment-based strategies such as Alignment Rajan et al. (2025), Dual

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



(a) Comparison with existing approaches.

(b) Ground-truth provenance labels

Figure 1: ForgeFormer jointly localizes provenance classes. Existing methods can fail to separate background and foreground in cross-manipulation settings such as hybrid edits.

Data Alignment (DDA) Chen et al. (2025b), and Bias-Free (BFree) training Guillaro et al. (2025) have proven effective in improving generalization across benchmarks. However, these methods generalize poorly to the diverse subtle manipulations found in the real-world partial fakes. While existing deepfake localization methods such as FakeShield Xu et al. (2025), SIDA, and SAFIRE perform poorly on both unseen full AI-generated images and detecting and localizing partial AI-edited images as well as physically edited images.

Building on these observations, we propose a large-scale, aligned benchmark, **ForgePan-5.9M**, for the detection and localization of manipulated images, comprising fully AI-generated, AI-edited, physically edited, or a combination of both physical and AI manipulations, termed "Hybrid Edit." As shown in Fig. 1a, existing methods such as SAFIRE fail to localize and separate unseen manipulations from the background, whereas ForgeFormer generalizes well across different kinds of manipulations and their combinations, cleanly disjoint from the background class. ForgeFormer is a panoptic segmentation-based approach trained on samples with constructed Ground Truth (GT) masks (Fig. 1b) from ForgePan-5.9M, achieving substantial performance gains on diverse commercial and non-commercial real-world OOD benchmarks and generalizing well to hybrid edits with forgeries in the foreground, compared to instance-level and binary-mask localization methods.

Our main contributions are threefold:

- We release 5,975,388 images in ForgePan-5.9M, comprising 995,898 samples in each of six supervised semantic classes (NATURAL, PHYSICAL EDIT, AI MASK EDIT, AI INPAINTING EDIT, FULLY AI-GENERATED, and HYBRID EDIT). The release includes 428,580 OOD test images in total; the remaining 5,546,808 images form the training and validation pool. The sources are OPA Liu et al. (2022a), MS-COCO Lin et al. (2014), VGG-Face2 Parkhi et al. (2015), OpenImages Kuznetsova et al. (2020), and FaceForensics++ Rossler et al. (2019). We additionally release a separate out-of-distribution evaluation corpus containing 1,790,712 synthetic images.
- We proposed ForgeFormer a panoptic segmentation-based approach consisting on DINOv3 vision encoder Siméoni et al. (2025) and a multi-scale feature pyramid with learnable forgery mask projection, which separates diverse manipulation types for generalizable forgery detection and localizing.
- ForgeFormer sets a new state-of-the-art localization pixel-level F1-score on four public benchmarks and is, to the best of our knowledge, the first model to jointly localize co-occurring AI and physical edits as well as classify them against natural and manipulated images in a single forward pass. It also closes the in-domain / out-of-domain gap observed in prior work: ForgeFormer trained on ForgePan-5.9M exceeds the closed-world baselines.

2 METHODOLOGY

The core methodology of ForgeFormer consists of three coupled stages: (i) an *operator-aligned data generation pipeline* (Sec. 2.1, Fig. 3) that produces ForgePan-5.9M by replaying the same mask geometry across five provenance branches so that the branches differ in the operator applied and not in where it is applied; (ii) the ForgeFormer *network* (Sec. 2.2, Fig. 6) that ingests an image and

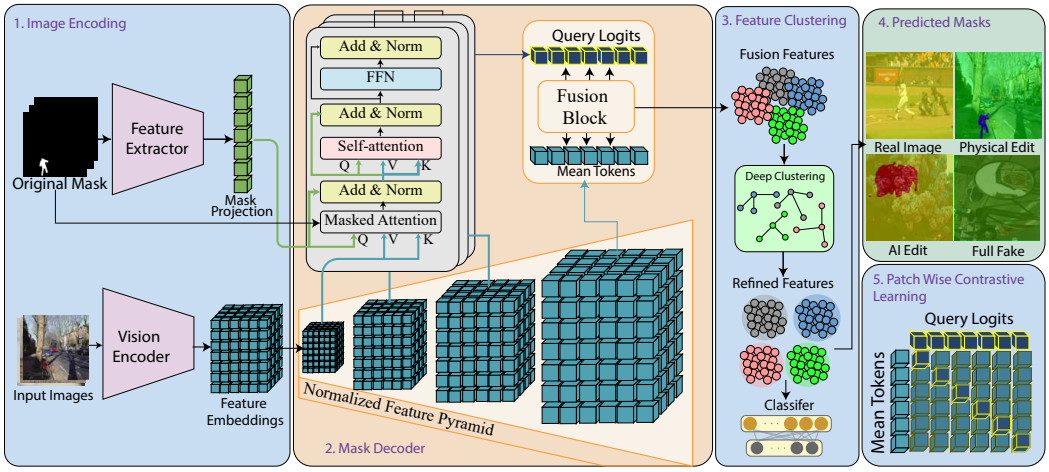


Figure 2: **ForgeFormer architecture.** (1) A LoRA-tuned DINOv3 encoder yields patch embeddings that, with a mask-projection token from the current mask estimate, feed (2) a Mask Decoder of N masked-cross-attention/self-attention/FFN blocks over a normalized multi-scale feature pyramid. (3) An online *Deep Clustering* (4) The decoder outputs a panoptic forgery map over five pixel-level classes. (5) A patch-wise supervised-contrastive head pulls same-provenance tokens in batch.

emits a six-way provenance label together with a panoptic forgery map; and (iii) a *training* protocol (Sec. 2.2.3) that combines a set-prediction loss with a patch-wise supervised contrastive objective and an unsupervised deep-clustering refinement of the latent feature space.

2.1 Data Generation Pipeline

A central failure mode of existing forgery benchmarks is the entanglement between *where* an edit lives and *what content* sits inside it: a handful of objects are repeatedly pasted, so detectors memorize *what* is edited instead of *that* something has been edited. Following the dataset-alignment view of Rajan et al. (2025); Chen et al. (2025b); Guillaro et al. (2025), we reuse the *same* mask geometry across all manipulation modalities *within a given editing turn* and replace the masked content with a different generator in each branch, so that branches differ only in the operator applied. The five aligned branches are $\{x_{\text{NAT}}, x_{\text{FULL}}, x_{\text{PHYS}}, x_{\text{MASK}}, x_{\text{INPAINT}}\}$. Alignment constrains *which operator* acts on a given region; it does not limit a sample to a single region, and a sample may receive several editing turns, each with its own mask.

Source pools. We sample from five public corpora. Scene images are drawn from **MS-COCO** Lin et al. (2014), **OpenImages** Kuznetsova et al. (2020) and **OPA** Liu et al. (2022a); face-centric images are from **VGG-Face2** Parkhi et al. (2015) and real frames of **FaceForensics++** Rossler et al. (2019).

Data Generation. Every input image is a pair (x, q), where x is source image and q is a category text query (Fig. 3a). For scene corpora, q is sampled from the object categories the corpus already annotates, such as "chair" or "woman"; for face corpora, q names a facial region such as "eyes", "lips", or "hair". Because q is drawn *before* any manipulation is applied, *where* an edit lands is decided independently of *what* is later placed inside it. We are precise about the scope of this: it decorrelates the mask from the *inserted* content, which is the confound that lets a detector learn what tends to be edited. It does not make the mask independent of the *source* scene, since q is drawn from the categories that source image already contains and the segmentation is conditioned on that image, so mask geometry remains correlated with scene composition. Appendix J states what this control does and does not remove. A language-guided segmentation model then turns (x, q) into a binary mask m (Fig. 3b; see Appendix E for the mask family $\mathcal{M}$), and that mask is replayed across the manipulation branches of the record, so that branches differ only in the operator that regenerates the masked pixels. Samples with more than one editing turn repeat this step per turn, drawing a fresh (q, m) each time.

Each input image expands into the five aligned branches introduced above. **(i) Natural.** $x_{\text{NAT}} = x$ is retained unmodified with an empty edit mask; it is the reference against which the other branches are aligned and supplies the negative provenance class for the patch-wise supervised contrastive

objective (Sec. 2.2.3). **(ii) AI Mask Edit.** x_{MASK} replaces the pixels under m with the corresponding region of a regenerated copy of x and blends the result back along the mask boundary. **(iii) AI Inpainting Edit.** x_{INPAINT} instead hands (x, m) to a diffusion inpainter under the identity-preserving instruction "regenerate the masked area as is", so the semantic content under m is preserved and only the generative fingerprint changes. Branches (ii) and (iii) share the editing stage of Fig. 3c but are kept as *distinct* semantic classes, because mask-based replacement and text-conditioned inpainting leave different traces at the mask boundary and in the regenerated interior. **(iv) Physical Edit.** x_{PHYS} applies a non-generative editor to the same mask (Fig. 3d): splice, copy-move, blending, or removal of the masked content. **(v) Fully AI-Generated.** x_{FULL} regenerates the entire image from x with a target generator (Fig. 3e); it is the one branch that carries no localized mask, since the manipulation covers the whole canvas and its panoptic label is assigned image-wide.

(vi) Hybrid edits. Hybrid Edit is the special case of multi-turn editing in which the turns mix a *non-generative* operator with a *generative* one, so that the same image contains both physically edited and AI-edited segments in spatially disjoint regions. Hybrid samples are constructed separately from the single-operator branches, but they form the sixth supervised *image-level* class and are used for training as well as for evaluation. Their panoptic annotation is not relabelled: each region keeps the class of the operator that produced it, so a hybrid sample supervises the *separation* of a physical from a generative edit in one image rather than collapsing both onto a single "hybrid" mask (Appendix D, Algorithm 2). Appendix E.1 gives the per-sample pseudocode, the generator inventory, and the sanity filters applied to every generated sample. ForgePan-5.9M contains $\mathbf{5{,}975{,}388}$ images: $\mathbf{995{,}898}$ samples in each of six supervised semantic classes (Natural, Physical Edit, AI Mask Edit, AI Inpainting Edit, Fully AI-Generated, and Hybrid Edit). Further details are given in Appendix. E.

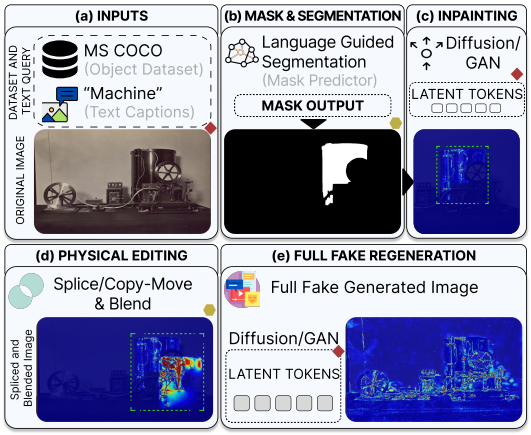


Figure 3: **Data Generation Pipeline. (a) Inputs:** A source image is drawn from open-source datasets. **(b) Segmentation & Masking:** the query and image are passed to a language-guided segmentation model (Lang-SAM, built on SAM Kirillov et al. (2023)) that emits a binary mask m. **(c) Inpainting/Editing:** the masked region is regenerated by diffusion or GAN-based models conditioned on the recovered category, producing separate AI MASK EDIT and AI INPAINTING EDIT samples according to the editing protocol. **(d) Physical Editing:** The same mask drives a non-generative splice/copy-move/blend pipeline. **(e) Full Fake Regeneration:** is a fully regenerated fake image from the source image using GAN or diffusion models.

2.2 FORGEFORMER ARCHITECTURE

Given an image $x \in \mathbb{R}^{3\times H\times W}$, ForgeFormer emits a set of Q mask class pairs $\{(\hat{m}_q, \hat{p}_q)\}_{q=1}^{Q}$ that are fused into (a) an image-level six-way label and (b) a per-pixel panoptic map. Five modules cooperate, in the order shown in Fig. 6: (1) image encoding, (2) mask decoder, (3) unsupervised deep-clustering refinement, (4) mask segmentation, and (5) patch-wise contrastive learning. Details of each component are given in the next sections.

2.2.1 VISION ENCODER

We use DINOv3-ViT-H+ Siméoni et al. (2025) with embedding dimension d=1280. Following the empirical observation of Yu et al. (2026) that self-supervised dense features transfer better to manipulation localization than ImageNet-supervised features, we keep the patch embedding and MLP blocks frozen and adapt only the self-attention projections via LoRA Hu et al. (2022) (rank 16). The encoder produces per-patch tokens $F_0 \in \mathbb{R}^{d\times \frac{H}{16}\times \frac{W}{16}}$.

Mask Projection. Let $M_q \in [0,1]^{H\times W}$ be the current estimate of the mask associated with query q. A multi-scale mask-projection M_q module converts into pyramid-aligned tokens $\tilde{M}_q$ that are added

to the object-query embeddings at each decoder layer. This initiates the "Mask Projection" block in Fig. 6; it lets the decoder condition cross-attention on the current mask prediction, producing a coarse-to-fine refinement. Beyond using predicted masks to gate attention, ForgeFormer projects each evolving mask into a learned query update; removing this feedback reduces mPQ^* from 6.2% and hybrid PQ from 10.66% (Table 3), supporting its contribution to provenance localization.

Normalized Feature Pyramid. We construct a three-scale pyramid $\{F_{1/32}, F_{1/16}, F_{1/8}\}$ with channel dimension d_m=256. $F_{1/16}$ is produced by a 1×1 projection of F_0; $F_{1/32}$ by a strided 3×3 convolution and $F_{1/8}$ by bilinear upsampling followed by a residual refinement block conditioned on a Laplacian-of-Gaussian band that exposes low-level boundary cues. Every level passes through a *feature-wise layer normalization* before fusion, which we find empirically critical to stabilize training when high-level DINOv3 features are mixed with low-level residual cues.

Mask Decoder. We stack N=9 decoder layers. Each layer consists of cross-attention, self-attention, and a feedforward network, cycling through the pyramid levels $\ell \in \{1/32, 1/16, 1/8\}$. Masked cross-attention attends only to positions when $\sigma(\hat{m}_q) > 0.5$, following Cheng et al. (2022a), where $\sigma(\cdot)$ represents a function to compute the probability of the manipulated region, while the first layer uses uniform attention. We use Q learned mask queries, each with a classifier over the five pixel-level provenance classes and a separate no-object output. Query count is a capacity setting, not the number of semantic classes; queries are matched to target regions by Hungarian assignment Cheng et al. (2022a). The pixel-level vocabulary is {NATURAL, FULLY AI-GENERATED, PHYSICAL EDIT, AI MASK EDIT, and AI INPAINTING EDIT}. HYBRID EDIT is deliberately absent from it: a hybrid image is one in which queries of *different* operator classes fire at once, so giving the decoder a sixth "hybrid" region label would let it satisfy the target by merging the two edits instead of resolving them. The hybrid verdict is therefore read off the image-level head of Sec. 2.2.3.

Segmentation Head. Each query emits (i) a class logit vector over the known classes plus a "no-object" class via a linear classifier, and (ii) a mask $\hat{m}_q \in [0,1]^{H\times W}$ via a dynamic convolution over $F_{1/8}$. The segmentation map is obtained by the standard Mask2Former fusion rule applied to the resulting set $\{(\hat{m}_q, \hat{p}_q)\}$: each pixel is assigned to the highest-scoring query whose binary mask covers it, ties broken by class confidence. This yields a single manipulation label per pixel matched with ground truth labels shown in Fig. 1b, enabling different colors for each manipulation to be interpretable for forensics analysis.

2.2.2 FEATURE CLUSTERING REFINEMENT VIA DEEP CLUSTERING

The supervised contrastive loss (Sec. 2.2.3) shapes the latent space using the known labels. However, within each class, the latent geometry is itself heterogeneous: different generators can leave distinct fingerprints within the same AI Mask Edit or AI Inpainting Edit class. We exploit this structure with an unsupervised refinement step similar to ***DeepCluster*** Caron et al. (2018): at regular intervals, we cluster the encoder features, use the resulting cluster indices as pseudo-labels, and treat them as auxiliary supervisory signals that pull each feature toward its assigned centroid.

Fusion Features. For each image, we form a fusion feature $u \in \mathbb{R}^{d_m}$ by concatenating the spatially average-pooled token of $F_{1/16}$ with the Q learned query embeddings Q emitted by the last decoder layer, followed by a two-layer MLP ϕ. The vector u is what flows into the "Fusion Block" of Fig. 6.

$$u = \phi\Big(F_{1/16}, Q\Big). \tag{1}$$

Online Clustering. At every T=500 training step we run a k-means pass with K=32 clusters over the running queue $\mathcal{U} = \{u_i\}_{i=1}^{B_\mathrm{q}}$ (queue size B_q=65,536), yielding centroids $\{c_k\}_{k=1}^{K}$ and pseudo-labels $\hat{z}_i \in \{1,\dots,K\}$. Following Caron et al. (2018), the centroid matrix is reinitialized whenever an empty cluster appears (we reassign the offending centroid to the largest cluster perturbed by Gaussian noise of variance 10^{-3}).

Cluster Loss. The refinement loss is a soft cross-entropy that pulls u_i towards its assigned centroid:

$$\mathcal{L}_\mathrm{DC} = -\frac{1}{|B|}\sum_{i\in B} \log \frac{\exp\big(u_i^\top c_{\hat{z}_i}/\tau_c\big)}{\sum_{k=1}^{K}\exp\big(u_i^\top c_k/\tau_c\big)}, \tag{2}$$

with τ_c=0.1. Because the ground-truth class labels are not used here, $\mathcal{L}_{\mathrm{DC}}$ acts as a regularizer on sub-class structure. Our intent is that it absorb recurring *manipulation patterns* mask family, generator, and blending mode to which the supervised loss is agnostic. We emphasize that this is a motivation and not a finding: the clustering operates on the image-level fusion vector u, we do not inspect or supervise what the K clusters encode, and they could equally track scene content or source corpus, both of which correlate with the generator in our release (Appendix J). The only evidence we offer is the ablation delta in Table 3; attributing it specifically to generator-level structure would require a cluster-purity analysis against held-out generator labels, which we do not run.

2.2.3 IMAGE-LEVEL CLASSIFICATION HEAD

A linear classifier $W_{\mathrm{cls}} \in \mathbb{R}^{6\times d_m}$ maps the fusion feature u to a six-way logit, $\hat{y} = \mathrm{softmax}(W_{\mathrm{cls}}u)$. The classifier is trained using label-smoothed cross-entropy (ϵ=0.15) with respect to the image-level provenance label. The six-dimensional softmax scores describe membership in the six supervised provenance classes, so the Hybrid Edit verdict is a supervised softmax logit rather than a rejection outcome. This head is the only place where class 5 is predicted: the decoder of Sec. 2.2.1 keeps the physical and the AI segments of a hybrid image apart, and the head reports, at image level, that both are present.

Patch-wise Supervised Contrastive Learning In parallel to the decoder, we apply a *patch-wise supervised-contrastive* loss Khosla et al. (2020) directly on individual patch tokens of $F_{1/16}$, each supervised by the provenance class of the *region it covers* rather than by the image label. Write $t_v \in \mathbb{R}^{d_m}$ for the token at spatial position v of $F_{1/16}$, and let $\Pi(v) \subset \Omega$ be the 16×16 pixel footprint that v subtends in the input. We project the ground-truth class map L of Appendix D onto the token grid under a purity rule: token v is assigned label $\ell(v) = c$ iff every pixel of $\Pi(v)$ carries class c, and is *discarded* otherwise. Discarding impure tokens removes exactly the footprints that straddle a segment boundary or intersect the 2-pixel void band, so no token is ever given a label that mixes two provenances, and the blending gradient at a mask edge is excluded from the objective instead of being averaged into it. To bound the pairwise cost we sample at most n_{tok} surviving tokens per image, stratified so that each class present in that image contributes equally; the index set I is the union of the sampled tokens over the batch. The loss is then

$$\mathcal{L}_{\mathrm{SupCon}} \;=\; \sum_{i\in I} \frac{-1}{|P(i)|} \sum_{p\in P(i)} \log \frac{\exp\big(t_i \cdot t_p/\tau\big)}{\sum_{a\in A(i)} \exp\big(t_i \cdot t_a/\tau\big)}, \tag{3}$$

with τ=0.07, where $P(i) = \{p \in I \setminus \{i\} : \ell(p) = \ell(i)\}$ is the positive set of tokens carrying the same *region* label as i, drawn from anywhere in the batch, and $A(i) = I \setminus \{i\}$. Positives and negatives are therefore formed at region granularity, not image granularity, which is what makes the objective meaningful on hybrid samples: a single hybrid image contributes PHYSICAL EDIT tokens and AI MASK/INPAINTING EDIT tokens to *different* positive sets, and its natural background to a third, so the loss pushes the two operator fingerprints apart within one image instead of collapsing them into a single pooled descriptor. Because the branches of Sec. 2.1 share mask geometry, tokens that differ only in the operator applied to the same region become explicit positive/negative pairs, which is the cue the objective is intended to isolate.

Training Objective. The total loss follows the Mask2Former set-prediction recipe augmented with our auxiliary terms:

$$\mathcal{L} \;=\; \lambda_{ce}\mathcal{L}_{ce} + \lambda_{dice}\mathcal{L}_{dice} + \lambda_b\mathcal{L}_{\mathrm{boundary}} + \lambda_c\mathcal{L}_{\mathrm{SupCon}} + \lambda_{\mathrm{dc}}\mathcal{L}_{\mathrm{DC}} + \lambda_{cls}\mathcal{L}_{\mathrm{cls}}, \tag{4}$$

with λ_{ce}=2.0, λ_{dice}=5.0, λ_b=1.0, λ_c=0.5, λ_{dc}=0.3, and λ_{cls}=2.0. $\mathcal{L}_{ce}$ is the query-to-class cross-entropy, $\mathcal{L}_{dice}$ is a generalized-Dice mask loss, $\mathcal{L}_{\mathrm{boundary}}$ is an L_1 loss between Sobel edges of the predicted and ground-truth masks, $\mathcal{L}_{\mathrm{cls}}$ is the image-level six-way cross-entropy with label smoothing 0.15 computed by the six-class image head, and $\mathcal{L}_{\mathrm{DC}}$ is the deep-clustering loss of Eq. equation 2. The assignment uses the Hungarian matcher of Cheng et al. (2022a), to match learned queries to the annotated regions of the five pixel-level classes present in the image; on a hybrid sample the targets are therefore a PHYSICAL EDIT region and an AI MASK/INPAINTING EDIT region, matched independently.

Table 1: **Per-class detection and localization on ForgePan-5.9M-test out-of-distribution (OOD).** Evaluation uses held-out generators (SD3, Flux.1-dev, DeepFillv2, and SD1.5), unseen physical-edit donor pools, and held-out OpenImages natural sources. AUC and Acc measure image-level detection; F1-score and IoU measure pixel-level localization. **best** and second-best.

Method	Natural				Physical Edit				AI Edit				Fully AI-Generated				Hybrid Edit			
	AUC	Acc	F1	IoU	AUC	Acc	F1	IoU	AUC	Acc	F1	IoU	AUC	Acc	F1	IoU	AUC	Acc	F1	IoU
MVSS-Net	49.22	48.75	13.44	12.36	69.83	63.45	40.56	38.74	62.54	49.86	32.43	31.65	59.73	55.67	15.34	13.45	44.82	43.34	12.23	11.74
PSCC-Net	51.11	49.43	21.65	19.87	75.56	68.75	48.83	38.66	69.76	54.65	34.54	31.75	65.71	63.21	40.99	37.83	58.28	50.33	18.74	14.56
TruFor	61.67	58.87	43.76	42.66	78.98	66.73	51.25	49.99	69.54	61.64	39.76	35.55	70.03	62.45	47.64	41.73	64.54	58.44	24.54	20.76
HiFi-Net	69.87	62.64	46.22	45.73	77.65	68.71	46.22	41.11	71.43	68.21	43.54	38.74	72.34	68.76	49.97	41.23	56.56	51.11	21.65	19.34
Mesorch	68.45	64.14	50.89	49.51	87.16	82.34	59.95	54.33	79.66	72.64	56.74	55.43	80.55	78.33	52.21	46.78	67.75	66.54	30.12	27.77
SIDA	80.17	78.45	60.29	57.34	91.12	87.28	60.42	58.29	85.22	81.61	69.34	65.31	88.72	82.02	65.43	61.14	68.23	67.83	40.54	39.88
RADAR	88.34	84.23	69.82	60.34	91.72	88.45	71.32	60.23	87.21	84.21	68.15	66.74	76.67	69.04	56.12	55.23	89.26	81.22	47.34	42.34
SAFIRE	92.23	89.99	79.23	74.32	94.23	92.34	82.34	80.34	90.34	89.96	84.44	80.22	89.31	88.64	79.23	78.45	90.23	89.23	65.25	64.22
ForgeFormer	**98.21**	**96.34**	**93.23**	**92.23**	**98.35**	**95.66**	**93.37**	**92.99**	**97.34**	**94.34**	**88.35**	**85.23**	**97.23**	**94.56**	**87.32**	**84.33**	**98.35**	**97.66**	**90.23**	**89.83**

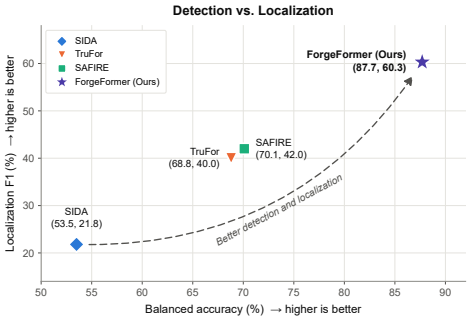


Figure 4: Performance comparison on the IEDAL2 challenge dataset.

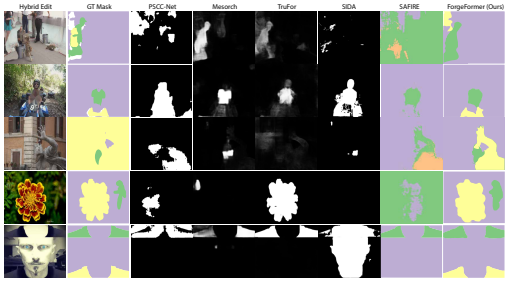


Figure 5: Qualitative comparison with different localization methods.

3 Experiments

3.1 Implementation details.

We train with AdamW, a peak learning rate 10^{-4} for the decoder and 2×10^{-5} for the LoRA adapters, weight decay 0.05, a cosine schedule, and a $1{,}500$-step linear warm-up. Images are processed at 512×512; we train for 24 epochs on 8 NVIDIA RTX PRO 6000 Blackwell GPUs with a global batch size of 64, using mixed precision (bf16), encoder activation, gradient checkpointing, and gradient clipping with a threshold of 1.0. Training takes approximately 4 days.

3.2 Evaluation Protocol

Benchmarks. We evaluate on four public benchmarks such as **CASIA v2** Dong et al. (2013), **Coverage** Wen et al. (2016), **IMD-2020** Novozámský et al. (2020), and **DiffSeg-30K** Ci et al. (2025) as well as our validation set, which consists of 1% samples from the training set, and a held-out test set consisting of multiple forgeries. We further evaluated the proposed approach on the Image Detection and Localization Challenge (IEDAL2) ULRI (2026) in which the evaluation is made on a private cloud without giving its access to the participants, and no training data is provided.

Metrics. For image-level verdicts, we report one-vs-rest accuracy and AUC over the six supervised semantic classes, of which Hybrid Edit is the sixth. Hybrid is a distinct manipulated class in this benchmark, not a synonym for "natural" or the decoder's no-object label. Localization is evaluated using pixel F1-score, IoU, and PQ over the five pixel-level classes, since a hybrid image is annotated with its constituent Physical Edit and AI Mask/Inpainting Edit segments rather than with a hybrid region label (Appendix. D). Further details are given in Appendix. B.2.

Baselines. We compare with **MVSS-Net** Dong et al. (2023), **PSCC-Net** Liu et al. (2022b), **CAT-Net** Kwon et al. (2021), **TruFor** Guillaro et al. (2023), **HiFi-Net** Guo et al. (2023), **Mesorch** Zhu et al. (2025), **SIDA** Huang et al. (2025), **RADAR** Costanzino et al. (2026), and **SAFIRE** Kwon et al. (2025). Further details about baselines are given in Appendix. B.1.

Table 2: **Per-generator detection accuracy (%).** Evaluation in terms of accuracy over 1,790,712 synthetic images from 19 commercial and non-commercial generators at the standard decision threshold of 0.5, together with 31,783 challenging real flicker samples (FLICKER30K) applied to every method for a fair comparison. Categories are talking-face reenactment (TF), end-to-end frame synthesis (EFS) and face swapping (FS); within each category, generators are ordered from worst to best under the proposed approach. **Bold** and underline mark the best and second-best results per row.

			Pre-trained on their own dataset				**Trained on ForgePan-5.9M**				
Cat.	**Generator**	**#Images**	SAFIRE	SIDA	TruFor	Radar	SAFIRE	SIDA	TruFor	Radar	**Ours**
TF	LatentSync Sync (2026)	21,296	41.45	32.33	44.21	17.73	58.30	49.88	68.82	26.91	**85.65**
	FLOAT Ki et al. (2025)	319,928	58.33	35.57	41.82	28.74	69.07	67.98	45.91	38.29	**96.86**
	Diff2Lip Mukhopadhyay et al. (2024)	21,040	49.82	47.89	40.34	30.72	56.82	51.81	68.38	36.33	**98.18**
	KDTalker Yang et al. (2025)	316,768	26.25	49.22	19.54	45.62	34.77	67.49	33.88	63.00	**99.14**
	MuseTalk Zhang et al. (2025b)	21,744	48.17	50.53	47.72	9.72	51.48	62.59	61.73	19.80	**96.42**
	VideoRetalking Cheng et al. (2022b)	19,536	40.77	19.82	18.78	5.01	41.86	30.42	29.62	18.81	**95.65**
	AniTalker Liu et al. (2024b)	319,952	5.83	2.93	7.82	32.74	9.91	5.77	11.61	55.34	**93.74**
EFS	LTX (I2V) HaCohen et al. (2024)	82,544	78.82	79.33	73.55	59.44	**98.51**	87.18	96.46	80.78	97.04
	WAN (I2V) Wan et al. (2025)	82,544	74.54	68.82	73.56	68.63	97.07	83.59	95.46	80.76	**97.64**
	LTX (T2V) HaCohen et al. (2024)	117,120	77.15	61.88	70.64	54.62	95.77	91.09	96.95	78.80	**97.49**
	Grok (T2V) xAI (2026a)	116,432	69.83	59.82	65.33	44.55	96.40	85.11	92.97	77.74	**99.04**
	WAN (T2V) Wan et al. (2025)	117,120	72.56	54.84	63.58	67.33	98.45	95.13	92.12	96.64	**99.89**
	FramePack (I2V) Zhang et al. (2025a)	82,544	88.23	82.48	76.72	41.57	99.37	94.02	98.28	88.72	**99.93**
	SkyReels (I2V) Chen et al. (2025a)	82,544	89.23	83.69	81.44	69.43	99.60	97.54	97.94	93.25	**100.00**
FS	Mango VideoSwap AI (2026)	15,360	68.11	33.58	66.73	48.72	90.35	49.11	81.93	89.87	**95.42**
	HyperSwap + CodeFormer FaceFusion (2026)	15,840	60.44	49.22	88.24	77.64	80.80	66.55	97.31	93.86	**98.85**
	SimSwap + RestoreFormer Chen et al. (2020)	15,360	48.82	31.75	79.72	57.62	66.34	55.67	93.93	92.65	**96.96**
	BlendFace + GFPGAN Shiohara et al. (2023)	11,520	42.33	29.74	58.22	73.52	63.80	47.56	**93.85**	87.48	92.00
	UniFace + GPEN Xu et al. (2022)	11,520	40.55	24.56	71.38	65.66	63.09	46.08	92.22	88.20	**96.00**
Fake accuracy (all generators)		1,790,712	49.87	44.72	44.08	44.19	62.74	63.91	58.83	64.96	**97.25**
Real accuracy (FLICKER30K Plummer et al. (2016))		31,783	50.66	39.72	57.21	38.12	49.38	69.04	79.29	43.20	**89.38**
Balanced Accuracy		**1,822,495**	50.26	42.22	50.64	41.15	56.06	66.48	69.06	54.08	**93.32**

3.3 Out of Distribution Performance

We compare the proposed approach with SAFIRE, SIDA, TruFor, and Radar on a large OOD test set spanning commercial and non-commercial generators (Table 2), where it outperforms the state of the art by a significant margin. We further evaluate it on the IEDAL challenge, which provides no training data and keeps the evaluation set hidden, so we train only on ForgePan-5.9M, with same setup and backbone of ForgeFormer and submit SIDA, TruFor, and SAFIRE to the same server for a blind, like-for-like comparison. The proposed approach ranks first on the leaderboard (Fig. 4), achieving 87.7% balanced accuracy and 60.3% localization F1-score, which is 17.6% and 18.3% above the strongest baseline. It also reaches 92.2% ROC AUC and 56.5% mean IoU, which is 15.0 and 18.5 points above the best baseline. Absolute localization scores are lower than on ForgePan-5.9M-test (Table 1) because the challenge generators and editing workflows are disjoint from our pipeline; the ranking, not the absolute value, is what generalizes and transfers. Details about the IEDAL2 challenge is given in Appendix F.

3.4 Qualitative Localization Performance

Figure. 5 shows artifact localization performance when compared with SAFIRE on OOD hybrid edited regions. Although Table 1 shows better localization performance on single forgeries, under hybrid edits, the method collapses and produces vague localization, despite being trained on ForgePan-5.9M. The proposed approach remains robust to OOD and produces sharp predictions that are very close to the Ground Truth (GT) masks. This is due to the panoptic nature of ForgeFormer and the fine-grained feature representations learned by the model.

3.5 Ablation Study

Table 3 presents findings on ForgeFormer evaluated on the full ForgePan-5.9M-test set. All variants are trained with the five pixel-level provenance classes and the six-way image head; PQ_{Hybrid} is read off the HYBRID EDIT subset, where the ground truth keeps the physical and the AI segments apart, so it rewards separating the two operators rather than merging them. The full model achieved 90.75 mPQ^{*} (equivalently 91.39 mPQ_{5c}) and 87.56 PQ_{Hybrid}, indicating the panoptic formulation's

Table 3: **Component ablation of ForgeFormer** on the full ForgePan-5.9M-test set. Each row removes, replaces, or scales one component; all variants use the five pixel-level provenance classes and the six-way image head. mPQ* is the mean of the five pixel-class PQs *and* the subset-level PQ_{Hybrid}. Since it is not class-averaged (nor is ΔmPQ*), these rows rank variants against each other, not against class-averaged PQ elsewhere. PQ_{Hybrid} is PQ on the HYBRID EDIT subset with physical and AI segments keeping their operator classes, measuring separation of co-occurring edits; $mIoU_{5c}$ averages the five pixel-level classes. Obj.: losses/objectives; Arch.: architecture; Cap.: capacity/scaling; BB: backbone. Δ: mPQ* change vs. the full model. **best**/second-best.

			Localization				Classification	
Cat.	**Variant**	**ΔmPQ***	mPQ* ↑	PQ_{Hybrid} ↑	$mIoU_{5c}$ ↑	Pixel-F1↑	Macro-$F1_{cls}$ ↑	AUC↑
Obj.	w/o patch-wise SupCon $\mathcal{L}_{\text{supcon}}$	−2.65	88.10	82.70	86.40	90.20	94.90	96.11
	w/o boundary edge loss $\mathcal{L}_{\text{boundary}}$	−1.75	89.00	84.10	87.50	91.10	95.40	97.22
	w/o DeepCluster refinement $\mathcal{L}_{\text{DC}}$	−3.15	87.60	80.90	85.80	89.70	94.30	93.26
	w/o image-level classification loss $\mathcal{L}_{\text{cls}}$	−2.05	88.70	83.20	86.90	90.80	92.60	–
	w/o Dice loss $\mathcal{L}_{\text{dice}}$	−3.85	86.90	79.80	84.50	88.60	94.10	95.42
	CE-only mask supervision	−8.35	82.40	74.10	80.20	85.10	91.80	93.15
Arch.	w/o mask projection	−6.25	84.50	76.90	81.60	86.80	93.10	96.99
	w/o normalized feature pyramid	−4.95	85.80	78.20	83.10	88.00	93.60	95.32
	w/o feature-wise layer normalization in NFP	−3.45	87.30	80.40	85.30	89.40	94.40	95.66
	w/o DINO–decoder feature fusion	−5.85	84.90	77.30	82.20	87.10	92.90	89.84
	w/o panoptic fusion (semantic segmentation only)	−9.95	80.80	70.60	78.40	83.70	91.20	96.89
	instance-level formulation (SAFIRE-style)	−11.85	78.90	66.80	76.90	82.40	90.70	97.11
	binary mask	−18.15	72.60	58.50	70.40	77.20	87.60	97.44
Cap.	decoder layers N: $9 \to 6$	−2.35	88.40	82.90	86.30	90.10	95.00	91.55
	decoder layers N: $9 \to 3$	−7.55	83.20	75.20	80.80	85.50	92.70	86.83
	input resolution $512 \to 256$	−4.55	86.20	79.40	83.80	88.30	94.00	92.78
	frozen DINOv3 encoder, no LoRA	−5.15	85.60	78.10	83.00	87.90	93.50	87.56
BB	DINOv3-H+ → DINOv2-L/14	−3.95	86.80	79.70	84.50	88.90	94.10	96.44
	DINOv3-H+ → CLIP-ViT-L/14	−9.05	81.70	72.90	79.10	84.20	91.80	71.45
	DINOv3-H+ → Swin-L (Mask2Former)	−11.15	79.60	69.40	77.20	82.50	90.60	65.53
	DINOv3-H+ → ResNet-101	−16.95	73.80	61.20	71.50	78.20	87.90	55.11
ForgeFormer (full model)		**0.00**	**90.75**	**87.56**	**89.90**	**93.40**	**96.80**	**97.86**

significance for multi-forgery localization. Replacing it decreases mPQ* substantially, with semantic segmentation dropping it by 9.95 and an instance-level formulation by 11.85. Hybrid edits showed even larger degradation. A binary mask approach performed poorly, with a drop of 18.15; because these rows hold data, supervision and budget fixed and vary only the output formulation, they are the comparison that isolates the formulation from the baselines' output-space handicap. Three architectural components are individually critical: removing mask projection, DINO-decoder feature fusion, or the normalized feature pyramid decreases mPQ* by 6.25, 5.85, and 4.95, respectively. Additionally, the ablation of SupCon, boundary loss, and others reveals their respective impacts on class separation and sharp contours. Lastly, scaling and backbone adjustments show that the gain is not merely a matter of capacity: halving the input resolution costs 4.55 mPQ* and reducing decoder depth from 9 to 3 costs 7.55, whereas enlarging the query set beyond Q=32 yields at most +0.87 mPQ* at two to four times the query cost (Table 8), and the choice of backbone directly affects mPQ*. Meanwhile, Swin-L represents Mask2Former architecture under exactly matched supervision reduces the performance by 11.15%, which is substantial.

4 CONCLUSION

We introduced **ForgePan-5.9M**, a large-scale, *operator-aligned* six-class panoptic forgery benchmark, and **ForgeFormer**, a Mask2Former-style architecture that turns forgery analysis from a binary heatmap problem into a panoptic provenance problem. Together they set a new state-of-the-art on four public benchmarks and on a held-out evaluation set whose images contain two or more co-occurring manipulation types. Our ablations, which hold data, supervision, and training budget fixed, attribute the absolute quality to the panoptic formulation, mask projection, and patch-wise contrastive learning. We stop short of the stronger claim that alignment is the *primary driver* of generalization: alignment removes one confound, several others remain in the released corpus, and a causal ranking would require a controlled aligned-versus-unaligned comparison matched on size, source composition, generator inventory, and post-processing, which we do not run.

ETHICS STATEMENT

ForgePan-5.9M is assembled from existing public research corpora and redistributes no new biometric data. OPA, MS-COCO, OpenImages, VGG-Face2, and FaceForensics++ are used under their own licenses, derived manipulations inherit the license of their parent image, and we honor the opt-out lists published by each source corpus (Section K). We release no face-identity labels beyond those the underlying corpora already distribute. Because a large share of the corpus is face-centric, its manipulated samples are synthetic depictions of real people; the image corpus is therefore distributed under a research-only license that forbids commercial use and forbids redistribution with identity labels.

Forgery detection is dual-use. A published detector also supplies an adversarial signal for evasion, and in moderation or chain-of-custody settings a confident but wrong provenance verdict can do more damage than no verdict at all. We address the first by documenting failure modes openly (Section H) rather than reporting headline numbers alone, and the second by reporting per-class and out-of-distribution behavior separately so that the classes on which ForgeFormer is weakest are visible to anyone deploying it rather than hidden inside an average. Section K.1 discusses the trade-off in full.

REPRODUCIBILITY STATEMENT

Everything needed to rebuild ForgePan-5.9M and retrain ForgeFormer is specified in the paper. Section I gives the complete hyperparameter manifest (Table 9): optimizer and the two learning rates, schedule and warm-up, batch size, input resolution, epoch count, LoRA rank, decoder depth N, query count Q, feature dimensions, and the loss weights of Equation (4), together with the hardware and wall-clock budget of the reported run. Section 2.2 specifies the encoder, the normalized feature pyramid, the mask-projection module, the decoder, and the two auxiliary objectives, and Section G reports the sweeps over λ_c, Q, and training-corpus size, so every reported setting can be located on its own curve rather than taken on trust.

The data pipeline is given as executable pseudocode rather than prose. Algorithm 1 generates one sample together with its image-level label, and Algorithm 2 converts the per-turn edit record into the panoptic annotation (L, S) of Section D. Section E documents the source corpora, the mask family $\mathcal{M}$, the identity-preserving prompt used for the inpainting branch, the generator inventory, the class-balancing procedure, and the per-sample sanity filters (mask consistency, LPIPS, and the pHash blocklist); Table 5 gives the per-source, per-generator and per-class counts of the released manifest. Source identities are disjoint across the training, validation, and out-of-distribution splits, and the pHash blocklist is applied against every evaluation set, so no evaluation image retains a counterpart in the training pool by construction.

The evaluation protocol is fixed before any result is reported. Section D defines the label space, the void bands, the thing/stuff assignment, and the panoptic quality of Equation (6), which is used as originally published and without modification. Section 3 names the benchmarks, the splits, the metrics, and the retraining protocol applied to every baseline, and Section L fixes the six post-processing perturbations and the single strength used for each. The challenge results of Section 3.3 are obtained on an evaluation set that participants never access, under a protocol we do not control.

Annotations are released in a form that can be checked rather than trusted. Each sample ships L and S as a two-channel PNG alongside a JSON record listing every segment, and the per-turn masks and the BUILDPANOPTIC implementation are released with them, so the segment decomposition behind every reported PQ value can be regenerated exactly instead of inferred from the class map. Section K states the licenses under which the code, the pretrained weights, and the image corpus are distributed. Source code and configuration files will be released.

GENAI USAGE DISCLOSURE

We used GenAI tools to assist with equation corrections and GPT-5.6 for PyTorch boilerplate code. AI usage was limited to improving presentation clarity and grammar checks.

REFERENCES

Mango AI. Ai video face swap, 2026. URL https://mangoanimate.com/ai/video-face-swap. Accessed: September 17, 2026.

Alibaba Cloud. Wan2.7-Image-Pro. Alibaba Cloud Model Studio documentation, 2026. URL https://www.alibabacloud.com/help/en/model-studio/wan2-7-image-pro. Accessed: 2026-09-25.

ByteDance Seed. Seedream 4.5. Official model page, 2025. URL https://seed.bytedance.com/en/seedream4_5. Accessed: 2026-09-25.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I*, pp. 213–229, Berlin, Heidelberg, 2020. Springer-Verlag. ISBN 978-3-030-58451-1. doi: 10.1007/978-3-030-58452-8_13. URL https://doi.org/10.1007/978-3-030-58452-8_13.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

Guibin Chen, Dixuan Lin, Jiangping Yang, Chunze Lin, Junchen Zhu, Mingyuan Fan, Hao Zhang, Sheng Chen, Zheng Chen, Chengcheng Ma, Weiming Xiong, Wei Wang, Nuo Pang, Kang Kang, Zhiheng Xu, Yuzhe Jin, Yupeng Liang, Yubing Song, Peng Zhao, Boyuan Xu, Di Qiu, Debang Li, Zhengcong Fei, Yang Li, and Yahui Zhou. Skyreels-v2: Infinite-length film generative model, 2025a. URL https://arxiv.org/abs/2504.13074.

Renwang Chen, Xuanhong Chen, Bingbing Ni, and Yanhao Ge. Simswap: An efficient framework for high fidelity face swapping. In *MM '20: The 28th ACM International Conference on Multimedia*, 2020.

Ruoxin Chen, Junwei Xi, Zhiyuan Yan, Ke-Yue Zhang, Shuang Wu, Jingyi Xie, Xu Chen, Lei Xu, Isabel Guan, Taiping Yao, and Shouhong Ding. Dual data alignment makes AI-generated image detector easier generalizable. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025b. URL https://openreview.net/forum?id=C39ShJwtD5.

Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA, 2021. Curran Associates Inc. ISBN 9781713845393.

Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1290–1299, June 2022a.

Kun Cheng, Xiaodong Cun, Yong Zhang, Menghan Xia, Fei Yin, Mingrui Zhu, Xuan Wang, Jue Wang, and Nannan Wang. Videoretalking: Audio-based lip synchronization for talking head video editing in the wild, 2022b.

Hai Ci, Ziheng Peng, Xuan Yingxin, Pei Yang, and Mike Zheng Shou. Diffseg30k: A multi-turn diffusion editing benchmark for localized AIGC detection, 2025. URL https://openreview.net/forum?id=wxoM2ky86C.

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 *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL https://openreview.net/forum?id=ciEdnQ0bLn.

Tiago José de Carvalho, Christian Riess, Elli Angelopoulou, Hélio Pedrini, and Anderson de Rezende Rocha. Exposing digital image forgeries by illumination color classification. *IEEE Transactions on Information Forensics and Security*, 8(7):1182–1194, 2013. doi: 10.1109/TIFS.2013.2265677.

Chengbo Dong, Xinru Chen, Ruohan Hu, Juan Cao, and Xirong Li. Mvss-net: Multi-view multi-scale supervised networks for image manipulation detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3539–3553, 2023. doi: 10.1109/TPAMI.2022.3180556.

Jing Dong, Wei Wang, and Tieniu Tan. Casia image tampering detection evaluation database. In *2013 IEEE China Summit and International Conference on Signal and Information Processing*, pp. 422–426, 2013. doi: 10.1109/ChinaSIP.2013.6625374.

FaceFusion. Hyperswap by facefusion labs, 2026. URL https://www.segmind.com/models/faceswap-hyperface-by-facefusion. Accessed: September 17, 2026.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 2021.

Google DeepMind. Nano Banana Pro (Gemini 3 Pro Image). Official model page, 2025. URL https://deepmind.google/models/gemini-image/pro/. Accessed: 2026-09-25.

Fabrizio Guillaro, Davide Cozzolino, Avneesh Sud, Nicholas Dufour, and Luisa Verdoliva. TruFor: Leveraging All-Round Clues for Trustworthy Image Forgery Detection and Localization . In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20606–20615, Los Alamitos, CA, USA, June 2023. IEEE Computer Society. doi: 10.1109/CVPR52729.2023.01974. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52729.2023.01974.

Fabrizio Guillaro, Giada Zingarini, Ben Usman, Avneesh Sud, Davide Cozzolino, and Luisa Verdoliva. A bias-free training paradigm for more general ai-generated image detection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 18685–18694, 2025.

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 (CVPR)*, pp. 3155–3165, June 2023.

Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, Poriya Panet, Sapir Weissbuch, Victor Kulikov, Yaki Bitterman, Zeev Melumian, and Ofir Bibi. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024.

HiDream.ai. HiDream-O1-Image. Official code and model repository, 2026. URL https://github.com/HiDream-ai/HiDream-O1-Image. Accessed: 2026-09-25.

Chih-Chung Hsu, Tzu-Yi Hung, Chia-Wen Lin, and Chiou-Ting Hsu. Video forgery detection using correlation of noise residue. In *2008 IEEE 10th Workshop on Multimedia Signal Processing*, pp. 170–174, 2008. doi: 10.1109/MMSP.2008.4665069.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR) 2025*, 2025.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 18661–18673. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/d89a66c7c80a29b1bdbab0f2a1a94af8-Paper.pdf.

Taekyung Ki, Dongchan Min, and Gyeongsu Chae. Float: Generative motion latent flow matching for audio-driven talking portrait. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 14699–14710, 2025.

Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation, 2019. URL https://arxiv.org/abs/1801.00868.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4015–4026, October 2023.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, 128(7):1956–1981, March 2020. ISSN 1573-1405. doi: 10.1007/s11263-020-01316-z. URL http://dx.doi.org/10.1007/s11263-020-01316-z.

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 *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 375–384, 2021. doi: 10.1109/WACV48630.2021.00042.

Myung-Joon Kwon, Wonjun Lee, Seung-Hun Nam, Minji Son, and Changick Kim. Safire: Segment any forged image region. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(4):4437–4445, Apr. 2025. doi: 10.1609/aaai.v39i4.32467. URL https://ojs.aaai.org/index.php/AAAI/article/view/32467.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), *Computer Vision – ECCV 2014*, pp. 740–755, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10602-1.

Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss (eds.), *Computer Vision – ECCV 2018*, pp. 89–105, Cham, 2018. Springer International Publishing. ISBN 978-3-030-01252-6.

Lei Liu, Peng Sun, Yubo Lang, and Jingjiao Li. Cfa-based splicing forgery localization method via statistical analysis. *IET Signal Processing*, 2024(1):9929900, 2024a. doi: https://doi.org/10.1049/2024/9929900. URL https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/2024/9929900.

Liu Liu, Zhenchen Liu, Bo Zhang, Jiangtong Li, Li Niu, Qingyang Liu, and Liqing Zhang. Opa: Object placement assessment dataset, 2022a. URL https://arxiv.org/abs/2107.01889.

Tao Liu, Feilong Chen, Shuai Fan, Chenpeng Du, Qi Chen, Xie Chen, and Kai Yu. Anitalker: Animate vivid and diverse talking faces through identity-decoupled facial motion encoding, 2024b. URL https://arxiv.org/abs/2405.03121.

Xiaohong Liu, Yaojie Liu, Jun Chen, and Xiaoming Liu. Pscc-net: Progressive spatio-channel correlation network for image manipulation detection and localization. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(11):7505–7517, 2022b. doi: 10.1109/TCSVT.2022.3189545.

Luma AI. UNI-1. Official technical specifications, 2026. URL https://lumalabs.ai/uni-1/tech-specs. Accessed: 2026-09-25.

Soumik Mukhopadhyay, Saksham Suri, Ravi Teja Gadde, and Abhinav Shrivastava. Diff2lip: Audio conditioned diffusion models for lip-synchronization. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 5292–5302, January 2024.

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

Alex 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, 2022. URL https://arxiv.org/abs/2112.10741.

Adam Novozámský, Babak Mahdian, and Stanislav Saic. Imd2020: A large-scale annotated dataset tailored for detecting manipulated images. In *2020 IEEE Winter Applications of Computer Vision Workshops (WACVW)*, pp. 71–80, 2020. doi: 10.1109/WACVW50321.2020.9096940.

Omkar M. Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. In *Proceedings of the British Machine Vision Conference (BMVC)*, pp. 41.1–41.12. BMVA Press, September 2015. ISBN 1-901725-53-7. doi: 10.5244/C.29.41. URL https://dx.doi.org/10.5244/C.29.41.

Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models, 2016. URL https://arxiv.org/abs/1505.04870.

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 *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=di52zR8xgf.

Qwen Team. Qwen-Image-Edit-2511. Official model card, 2025. URL https://huggingface.co/Qwen/Qwen-Image-Edit-2511. Accessed: 2026-09-25.

Anirudh Sundara Rajan, Utkarsh Ojha, Jedidiah Schloesser, and Yong Jae Lee. Aligned datasets improve detection of latent diffusion-generated images. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=doBkiqESYq.

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 (CVPR)*, pp. 10684–10695, June 2022a.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Bjorn Ommer. High-Resolution Image Synthesis with Latent Diffusion Models . In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10674–10685, Los Alamitos, CA, USA, June 2022b. IEEE Computer Society. doi: 10.1109/CVPR52688.2022.01042. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52688.2022.01042.

Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Niessner. FaceForensics++: Learning to Detect Manipulated Facial Images . In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1–11, Los Alamitos, CA, USA, November 2019. IEEE Computer Society. doi: 10.1109/ICCV.2019.00009. URL https://doi.ieeecomputersociety.org/10.1109/ICCV.2019.00009.

Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, SIGGRAPH '22, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450393379. doi: 10.1145/3528233.3530757. URL https://doi.org/10.1145/3528233.3530757.

Kaede Shiohara, Xingchao Yang, and Takafumi Taketomi. Blendface: Re-designing identity encoders for face-swapping. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7634–7644, October 2023.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. Dinov3, 2025. URL https://arxiv.org/abs/2508.10104.

Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 3172–3182, 2022. doi: 10.1109/WACV51458.2022.00323.

Latent Sync. Latent sync: Lip sync free, 2026. URL https://www.latentsync.org/. Accessed: September 17, 2026.

Eren Tahir and Mert Bal. Deep image composition meets image forgery, 2024. URL https://doi.org/10.21203/rs.3.rs-4744410/v1.

Tencent. HunyuanImage-3.0. Official model card, 2025. URL https://huggingface.co/tencent/HunyuanImage-3.0. Accessed: 2026-09-25.

Vrizlynn L.L. Thing, Yu Chen, and Carmen Cheh. An improved double compression detection method for jpeg image forensics. In *2012 IEEE International Symposium on Multimedia*, pp. 290–297, 2012. doi: 10.1109/ISM.2012.61.

ULRI. Iedal2: Image edit detection and localization challenge (part 2). https://app.dyff.io/challenges/7933ea10ee0f4a3999c85cd1dca502a1?tab=leaderboard&task=be32157a5f7649a791260b1f4438a395, 2026. Accessed: September 17, 2026.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models, 2025. URL https://arxiv.org/abs/2503.20314.

Bihan Wen, Ye Zhu, Ramanathan Subramanian, Tian-Tsong Ng, Xuanjing Shen, and Stefan Winkler. Coverage — a novel database for copy-move forgery detection. In *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 161–165, 2016. doi: 10.1109/ICIP.2016.7532339.

Biao Wu, Tengfei Cheng, Meng Fang, and Ling Chen. Can large vision-language models refuse synthetic images in geo-localization?, 2025. URL https://openreview.net/forum?id=1BGD0eRkik.

Yue Wu, Wael Abd-Almageed, and Prem Natarajan. Busternet: Detecting copy-move image forgery with source/target localization. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss (eds.), *Computer Vision – ECCV 2018*, pp. 170–186, Cham, 2018. Springer International Publishing. ISBN 978-3-030-01231-1.

Yue Wu, Wael AbdAlmageed, and Premkumar Natarajan. Mantra-net: Manipulation tracing network for detection and localization of image forgeries with anomalous features. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9535–9544, 2019. doi: 10.1109/CVPR.2019.00977.

xAI. Grok imagine video (version 1.5), 2026a. URL https://docs.x.ai/developers/model-capabilities/video/generation. Text-to-Video and Image-to-Video foundation model with native audio.

xAI. Grok Imagine: Image Generation. Official API documentation, 2026b. URL https://docs.x.ai/developers/model-capabilities/images/generation. Accessed: 2026-09-25.

Chao Xu, Jiangning Zhang, Yue Han, Guanzhong Tian, Xianfang Zeng, Ying Tai, Yabiao Wang, Chengjie Wang, and Yong Liu. Designing one unified framework for high-fidelity face reenactment and swapping. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XV*, pp. 54–71, Berlin, Heidelberg, 2022. Springer-Verlag. ISBN 978-3-031-19783-3. doi: 10.1007/978-3-031-19784-0_4. URL https://doi.org/10.1007/978-3-031-19784-0_4.

Zhipei Xu, Xuanyu Zhang, Runyi Li, Zecheng Tang, Qing Huang, and Jian Zhang. Fakeshield: Explainable image forgery detection and localization via multi-modal large language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=pAQzEY7M03.

Chaolong Yang, Yinuo Guo, Kai Yao, Yuyao Yan, Jie Sun, and Kaizhu Huang. Kdtalker++: Controllable talking portrait generation with audio, text, and expression editing. In *Proceedings of the 33rd ACM International Conference on Multimedia*, MM '25, pp. 13486–13488, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400720352. doi: 10.1145/3746027.3754462. URL https://doi.org/10.1145/3746027.3754462.

Jieming Yu, Qiuxiao Feng, Zhuohan Wang, and Xiaochen Ma. Dinov3 beats specialized detectors: A simple foundation model baseline for image forensics, 2026. URL https://arxiv.org/abs/2604.16083.

Lingzhi Zhang, Tarmily Wen, and Jianbo Shi. Deep image blending, 2019. URL https://arxiv.org/abs/1910.11495.

Lvmin Zhang, Shengqu Cai, Muyang Li, Gordon Wetzstein, and Maneesh Agrawala. Frame context packing and drift prevention in next-frame-prediction video diffusion models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025a. URL https://openreview.net/forum?id=J8JCF64aEn.

Yue Zhang, Zhizhou Zhong, Minhao Liu, Zhaokang Chen, Bin Wu, Yubin Zeng, Chao Zhan, Yingjie He, Junxin Huang, and Wenjiang Zhou. Musetalk: Real-time high-fidelity video dubbing via spatio-temporal sampling, 2025b. URL https://arxiv.org/abs/2410.10122.

Xuekang Zhu, Xiaochen Ma, Lei Su, Zhuohang Jiang, Bo Du, Xiwen Wang, Zeyu Lei, Wentao Feng, Chi-Man Pun, and Ji-Zhe Zhou. Mesoscopic insights: Orchestrating multi-scale hybrid architecture for image manipulation localization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(10):11022–11030, Apr. 2025. doi: 10.1609/aaai.v39i10.33198. URL https://ojs.aaai.org/index.php/AAAI/article/view/33198.

APPENDIX

A TECHNICAL APPENDICES AND SUPPLEMENTARY MATERIAL

This appendix provides the full documentation for the ForgePan-5.9M benchmark (Sec. Section E), implementation details of the data generation pipeline (Sec. E.1), additional ablations (Sec. G), failure-mode analysis (Sec. H), hyperparameters (Sec. I), the scope of the alignment claim and the confounds it leaves open (Sec. J), (Sec. K), broader impact (Sec. K.1), and robustness to post-processing (Sec. L).

B ADDITIONAL EXPERIMENTAL SETUP

B.1 BASELINE SETUP

Baselines that emit only a binary forgery mask cannot express a typed segment, so we adapt them with an explicit, uniform conversion rather than comparing against them directly. Each baseline keeps its own localization output. We threshold it at 0.5, extract 8-connected components, and discard components below 128 pixels, matching the filter applied to our own predictions (Sec. 2.2.1). Every surviving component becomes a candidate segment, is assigned its own identifier, and is typed by a region classifier: the baseline's last feature map is average-pooled over the component's support and passed to a linear head over the five pixel-level provenance classes, trained on ForgePan-5.9M with the same schedule and budget as the baseline itself. Where two components overlap, the pixel is awarded to the higher region-classifier confidence, so each pixel receives exactly one label and the output is a valid panoptic map. The image-level verdict comes from a separate six-way head trained on the same features.

We stress what this comparison does and does not establish. The conversion is an adaptation *we* impose, and a binary-output model evaluated against a class-aware panoptic target is partly being measured on an output capability it was never trained to have; the class-aware margins in Tables 4 and 3 therefore confound representation quality with output-space expressiveness and should not on their own be read as evidence that ForgeFormer learns stronger forensic features. The architecture-matched comparisons in Table 3 are the relevant controls, since they hold data, adaptation, supervision, and training budget fixed while varying the output formulation and the backbone. Table 2 presents frame-level accuracy of around 223,839 fake videos with 8 frames extracted from each video. We extracted frames and reported frame-level accuracy because the proposed approach is an image-based detector and there is no temporal information being processed. However, the given samples represent minor edited regions, and a fully generated region exists in real-world scenarios. The proposed approach shows better generalization to unseen commercial and non-commercial generators both in Table 2 and the IEDAL2 challenge, which contains commercial tools such as Nano Banana Pro, GPT-Image 2.5, and Grok Imagine.

B.2 ADDITIONAL METRICS DETAILS

We additionally report $\mathrm{PQ}_{\mathrm{Hybrid}}$, the panoptic quality restricted to the HYBRID EDIT subset: because the operator classes are retained there, this number measures whether a method resolves a physical and an AI edit *as different classes within one image*, which is the capability claimed in Sec. 1, and not merely whether it finds the edited area.

Two summaries are reported and are *not* interchangeable. mPQ_{5c} is the class-averaged panoptic quality of Eq. equation 6, the unweighted mean over the five pixel-level classes, and the quantity that matches PQ as originally defined. mPQ^{*} is a composite that averages those five class-level scores together with the subset-level $\mathrm{PQ}_{\mathrm{Hybrid}}$; mixing a class-level with a subset-level term makes it a custom measure rather than a class-averaged one, and we flag it as such wherever it appears. Our ablations and robustness tables are indexed by mPQ^{*} for continuity with the configuration search, while Table 4 reports both. Unless stated otherwise, every panoptic number in this paper is computed on the full ForgePan-5.9M-test set; the HYBRID EDIT subset is used only for $\mathrm{PQ}_{\mathrm{Hybrid}}$, and the multi-forgery split of Appendix E is released but not used as an evaluation set here.

C Related Work

C.1 Manipulation-Specific Detectors

Existing forgery detectors target low-level artifacts of conventional editing, color-filter-array inconsistency, JPEG double compression, and sensor-noise residuals Liu et al. (2024a); Thing et al. (2012); Hsu et al. (2008), and while they cannot distinguish AIGC where no camera-level fingerprint exists. Learned successors widen the feature space but not the output space: MVSS-Net's dual RGB/noise stream Dong et al. (2023), PSCC-Net's spatio-channel correlation blocks Liu et al. (2022b), CAT-Net's compression-artifact tracking Kwon et al. (2021), BusterNet's copy-move split Wu et al. (2018), and ManTraNet's anomalous-feature encoder Wu et al. (2019) these methods produce a binary foreground mask highlighting a single fake region. TruFor Guillaro et al. (2023) adds a reliability map atop a learned camera fingerprint but keeps the same collapse Guo et al. (2023) global-local hierarchy, FakeShield's multimodal-LLM explanations Xu et al. (2025), and the mesoscopic hybrid architecture of Zhu et al. (2025) all still resolve to highlighting one foreground fake region per image. RADAR Costanzino et al. (2026) tunes a reliability-weighted head specifically for diffusion artifacts; SIDA Huang et al. (2025) targets social-media deepfake detection and explanation Wu et al. (2025) probe whether frontier VLMs localize fine-grained AI edits Tahir & Bal (2024) build a composition-based splice dataset. These report strong numbers within their own manipulation type, but precisely because they specialize on generator-specific fingerprints, they generalize poorly once an image contains a different manipulation type, a physical edit, a full synthesis, or a mixture and vice versa: physical-edit specialists collapse on AI-edited content. The most recent efforts move past binary masks by segmenting multiple instances within one image but stop short of representing manipulation type. SAFIRE Kwon et al. (2025) and DiffSeg-30K Ci et al. (2025) segment and classify regions by generator-specific artifacts, a genuine advance over a black and white foreground mask, but their label vocabulary is generator identity, not manipulation category. This conflates the identity of the method with the nature of the manipulation: two regions produced by different diffusion checkpoints are given different labels even when both are, semantically, "AI-edited," while an image's physically edited or fully synthesized regions have no representation in this vocabulary at all. Consequently, both methods degrade substantially when evaluated on heterogeneous categories natural, physically edited, AI-edited, fully synthesized, and hybrid that lie outside their generator-indexed training distribution, limiting their generalization ability to better distinguish forgery regions from background.

C.2 Aligned Data and the Segmentation Backbone

The weaknesses above are compounded at the data level. Widely used benchmarks IMD-2020 Novozámský et al. (2020), DSO-1 de Carvalho et al. (2013) and Coverage Wen et al. (2016) reuse a small, repeatedly-pasted gallery of objects, so a detector can learn what tends to be edited rather than that something was edited: a shortcut that inflates in-domain accuracy while destroying out-of-distribution generalization. Alignment-based training Rajan et al. (2025), Chen et al. (2025b), Guillaro et al. (2025) addresses this by pairing a natural image with matched manipulated counterparts under controlled conditions, forcing a model onto forgery-relevant cues rather than scene-content shortcuts. However, each of these efforts aligns only one manipulation type against natural e.g., full AI generation vs. natural, or a single edit type vs. natural. None aligns physical edits, AI edits, and full synthesis simultaneously against the same natural image over the same mask geometry, which is what is needed to isolate the manipulation fingerprint from every other confound at once, across every manipulation type a detector must eventually generalize across. On the architecture side, we build on the set-prediction segmentation template established by Carion et al. (2020), MaskFormer Cheng et al. (2021) and Mask2Former Cheng et al. (2022a), and on the observation of Yu et al. (2026) that DINOv3 features adapted via LoRA Hu et al. (2022) transfer better than ImageNet-supervised backbones to manipulation localization; M2SFormer Nam et al. (2025) similarly shows gains from unifying multi-frequency, multi-scale cues for binary localization. None of this architecture lineage, however, has been applied to a provenance-panoptic formulation, or paired with a dataset aligned across manipulation types rather than against natural alone the gap ForgePan-5.9M and ForgeFormer are built to close.

C.3 PANOPTIC SEGMENTATION PERFORMANCE

Table 4 reports panoptic quality over the five pixel-level provenance classes, plus a sixth column, HYB., which is the panoptic quality restricted to the HYBRID EDIT split, i.e. the images whose image-level label is 5 and whose annotation therefore contains physical *and* AI segments at once. Because those segments keep their operator classes (Sec. D), the HYB. column scores whether a method separates the two operators in a single image rather than whether it finds an undifferentiated edited area. The proposed approach achieved better panoptic segmentation performance, as the method is well suited for panoptic segmentation and better separates background from the foreground manipulations. The second best model is SAFIRE, which achieves better performance due to its instance-level segmentation approach.

D TASK FORMULATION

Label space. We use a closed vocabulary of six provenance classes,

$$\mathcal{C} = \{\, 0{:}\text{NATURAL},\ 1{:}\text{FULLY AI-GENERATED},\ 2{:}\text{PHYSICAL EDIT},\\ 3{:}\text{AI MASK EDIT},\ 4{:}\text{AI INPAINTING EDIT},\ 5{:}\text{HYBRID EDIT}\,\},$$

together with a reserved *void* label $\varnothing$ that is excluded from both the training targets and the metric. The vocabulary is used at two granularities. All six classes are *image-level* verdicts. At the *pixel* level only classes 0–4 are ever written into L: HYBRID EDIT is by construction a statement about an *image* that carries segments of more than one operator class, so painting those segments with a single class 5 would erase precisely the distinction the class exists to report. A hybrid image is therefore annotated with PHYSICAL EDIT and AI MASK/INPAINTING EDIT segments side by side in L, and carries an image-level label 5. This is what makes the localization claim of Sec. 1 measurable: recovering the two operators *as different classes* in one image is scored, not assumed. NATURAL is a scored class denoting authentic content, *not* the absence of an annotation; void is a separate value used only for the ambiguity bands defined below. Keeping the two distinct matters, because NATURAL is reported as a scored class in Table 4.

Thing/stuff assignment. NATURAL and FULLY AI-GENERATED are *stuff*: each contributes at most one segment per image. NATURAL covers the unedited complement of the edited regions, and FULLY AI-GENERATED covers the entire canvas, since that branch carries no localized mask. PHYSICAL EDIT, AI MASK EDIT and AI INPAINTING EDIT are *things*: every connected component of every edited region is its own segment with its own identifier, and it keeps that operator class whether or not other operators are present elsewhere in the image. HYBRID EDIT receives no thing/stuff assignment, because it is not a pixel class.

Construction of (L, S). Let $\{(m_j, c_j)\}_{j=1}^{J}$ be the masks and class labels deposited by the J editing turns applied to a sample be in turn ordered. Edits are composited in that order, so a later edit overwrites an earlier one wherever they overlap and every pixel ends up with exactly one provenance label. With $\mathrm{CC}(\cdot)$ denoting 8-connected component extraction, the segment set is

$$\mathcal{S} = \underbrace{\Big\{(r,\, c_j)\ :\ r \in \mathrm{CC}\big(m_j \setminus \textstyle\bigcup_{j'>j} m_{j'}\big),\ j = 1, \ldots, J\Big\}}_{\textit{thing}\text{ segments, one per connected component}} \cup \Big\{\big(\Omega \setminus \textstyle\bigcup_j m_j,\ \text{NATURAL}\big)\Big\}, \quad (5)$$

and S numbers the elements of $\mathcal{S}$ consecutively. A single-edit sample ($J{=}1$ with one connected component) reduces to one edited segment plus one NATURAL segment; the FULLY AI-GENERATED branch is the degenerate case $L \equiv 1$, $S \equiv 1$. Two regions are set to void and excluded from scoring: connected components smaller than 128 pixels, matching the filter applied to predictions at inference (Sec. 2.2.1), and a 2-pixel band on either side of each segment boundary, so that PQ is not dominated by sub-pixel blending ambiguity at mask edges. We release L and S as two channels of a single PNG per image, together with the deterministic script that regenerates them from the released masks, so the segment decomposition underlying every reported number can be reproduced exactly.

Panoptic Quality. We use PQ as originally defined (Kirillov et al., 2019), with no modification. A predicted segment p and a ground-truth segment g *match* iff they carry the same class and $\mathrm{IoU}(p, g) > 0.5$; this threshold makes the matching unique, so no greedy assignment is needed. Per

class c,

$$\mathrm{PQ}_c = \frac{\sum_{(p,g)\in \mathrm{TP}_c} \mathrm{IoU}(p,g)}{|\mathrm{TP}_c| + \frac{1}{2}|\mathrm{FP}_c| + \frac{1}{2}|\mathrm{FN}_c|}, \tag{6}$$

where TP_c, FP_c and FN_c are the matched, unmatched-predicted, and unmatched-ground-truth segments of class c. Void pixels are removed from both arguments of every IoU, and a predicted segment whose overlap with void exceeds 50% of its area is discarded before matching rather than counted as a false positive. mPQ_{5c} is the unweighted mean of PQ_c over the five pixel-level classes, and is the only quantity in this paper that is a class-averaged PQ in the sense of Kirillov et al. (2019). The composite mPQ^* used to index the ablations additionally averages in $\mathrm{PQ}_{\text{Hybrid}}$, a *subset*-level score computed over the HYBRID EDIT images; mixing a subset-level with class-level terms makes it a custom summary, and we label it as such wherever it is reported.

E DATA GENERATION PROCESS AND DATASHEET FOR FORGEPAN-5.9M

AI Mask Edit and AI Inpainting Edit We derive a binary mask $m \in \{0,1\}^{H\times W}$ using the language-guided segmentation model GroundingDINO, which combines a CLIP-style text encoder with SAM. Unlike fixed bounding boxes or instance annotation sampling, GroundingDINO produces masks closely aligned with object boundaries and is independent of any pretrained object identity biases. To expand the mask family $\mathcal{M}$, we sample from various methods (GroundingDINO, elliptic, Voronoi, free-form strokes, 68-landmark convex hull, axis-aligned rectangle) and uniformly sample mask areas on a log scale. Mask generation occurs before the editing phase, which makes the mask location independent of the content later placed inside it. It does not make the mask independent of the source scene, since the query is drawn from the categories that image already contains; see Sec. J. Using the source image, mask, and scene corpora, we regenerate masked regions through diverse diffusion methods (e.g., SDXL-Turbo, SDXL3.5) and GAN-based techniques (e.g., edgeconnect, Deepfillv2). We also create inpainting images with an identity-preserving prompt $\pi(x, m, q)$, which biases the process to regenerate semantically consistent content. These two protocols define distinct semantic classes: AI Mask Edit uses mask-based editing, whereas AI Inpainting Edit uses the identity-preserving text-conditioned inpainting protocol. Both classes have equal release-wide counts and reuse the sampled mask geometry; GAN-based editing uses only the mask. Dataset balance was achieved using SDXL-Turbo.

Physical Edit Using the same mask m, we apply a non-generative editor sampled uniformly from {splice, copy-move, face-swap (alignment-only), poisson blend Zhang et al. (2019), and copy-paste}. For splice and copy-move, we harvest a donor patch from a class-matched image, warp it using thin-plate splines, and composite it using Poisson blending, automatic color transfer, and a final JPEG recompression with $Q \sim \mathcal{U}(75, 95)$. The panoptic label inside m is set to class 2; outside m, the image remains natural. Given (x, m, m_2, q), containing two masks for diffusion-based AI inpainting (m), as well as for physical editing (m_2), the mask region m is regenerated by different diffusion-based inpainting methods, while m_2 is edited with copy-move, splice, or remove as given in the previous stage, with similar conditions as stated before. We assign class IDs 0–5 to Natural, Fully AI-Generated, Physical Edit, AI Mask Edit, AI Inpainting Edit, and Hybrid Edit, respectively. Hybrid images combine physical edits with AI mask edits or AI inpainting edits. Their image-level ground-truth label is Hybrid Edit (ID 5), rather than Natural, and the supervised classifier is trained on Hybrid samples. Their *panoptic* map, by contrast, carries no class 5: each edited region keeps the class of the operator that produced it (2, 3, or 4), so that a hybrid image supervises the separation of physical from AI edits instead of merging them.

Multi-turn editing and the multi-forgery split. The five branches name the manipulation *operator*; they do not restrict a sample to a single edited region. A sample is produced by applying $J \geq 1$ editing *turns*, each of which draws a fresh mask from the mask family $\mathcal{M}$ and an operator and composites its result onto the output of the previous turn. Two consequences are what make the task panoptic rather than semantic. First, a single image routinely carries *several disjoint segments of the same provenance class* for example two independently spliced regions which a semantic class map cannot distinguish but which Eq. equation 5 separates by identifier. Second, different turns may draw different operators, so one image can carry segments of *different* provenance classes at once. We additionally define and release a *multi-forgery* split, the subset of the test pool whose samples satisfy $J \geq 2$, i.e. images containing two or more distinct manipulated segments. It is provided so that

multi-region behavior can be isolated, but every panoptic number reported in this paper (Tables 4, 3 and 11) is computed on the *full* ForgePan-5.9M-test set, not on this subset; the HYBRID EDIT subset used for PQ_{Hybrid} is a different and narrower selection, namely the images whose image-level label is 5.

Fully AI-Generated We utilized a prompt, x ="*Regenerate the same image as without changing anything.* " is used to condition the diffusion model to regenerate the entire image, $x_{\text{FULL}} = \text{Generator}(x)$ whereas $Generator$ refers to diffusion or GAN-based generators. This strategy generates similar fake images while retaining the fingerprints needed to generalize to OOD datasets, thereby reducing bias, and the label for the full image is given as a panoptic label. For LaMa, DeepFillv2, and EdgeConnect, which are designed primarily for image inpainting rather than native full-image generation, we divide the image into regions whose union covers the complete image and regenerate every region using the corresponding model. Thus, all pixels in the resulting image have been processed by the inpainting model, allowing its characteristic forensic fingerprints to extend across the entire image; these samples are therefore also assigned the FULLY AI-GENERATED label. We adopt the datasheet format of Gebru et al. (2021).

Table 4: Panoptic quality (PQ) on the full ForgePan-5.9M-test set. The first five columns are per-class PQ over the pixel-level vocabulary: Nat.: natural; Phys.: physical edit; AI-M.: AI mask edit; AI-Inp.: AI inpainting edit; Full: fully AI-generated. Hyb. is *not* a sixth pixel class but the PQ restricted to the HYBRID EDIT subset, whose images carry a physical *and* an AI segment that keep their operator classes (Appendix D); it therefore scores separating co-occurring edits, not merely detecting them. The two summary columns are reported separately and must not be conflated. $\mathbf{mPQ}_{5c}$ is the class-averaged panoptic quality of Equation (6), the unweighted mean of the five pixel-class entries; it is the quantity comparable to PQ as published. $\mathbf{mPQ}^{*}$ is a *composite* that averages those five class-level scores together with the one subset-level score in the Hyb. column and is therefore not a class-averaged metric. We report mPQ^{*} because it is the summary used by the ablations of Table 3, and mPQ_{5c} because it is the one that matches the definition in Section D. **Bold** and underline mark the best and second-best results.

Method	PQ per pixel class↑					Subset↑	Summary↑	
	Nat.	Phys.	AI-M.	AI-Inp.	Full	Hyb.	mPQ^{*}	mPQ_{5c}
MVSS-Net	31.25	45.26	41.98	40.88	39.91	34.76	39.01	39.86
TruFor	35.67	52.56	47.98	45.67	51.65	36.67	45.03	46.71
SIDA	42.65	59.98	62.65	60.21	41.78	37.89	50.86	53.45
RADAR	48.65	68.95	78.36	<u>77.98</u>	58.76	52.56	64.21	66.54
SAFIRE	<u>68.88</u>	<u>84.78</u>	<u>82.94</u>	<u>77.98</u>	<u>62.70</u>	<u>58.85</u>	<u>72.69</u>	<u>75.46</u>
ForgeFormer	**92.20**	**94.49**	**89.54**	**85.85**	**94.88**	**87.56**	**90.75**	**91.39**

Motivation. ForgePan-5.9M was created to enable a *provenance-panoptic* formulation of image forgery analysis, addressing two failures of existing datasets: (i) tiny class vocabularies that collapse all manipulations into one "fake" bucket, and (ii) a strong correlation between the edited region and scene semantics that lets detectors memorize content.

Motivation. ForgePan-5.9M was created to enable a *provenance-panoptic* formulation of image forgery analysis, addressing two failures of existing datasets: (i) tiny class vocabularies that collapse all manipulations into one "fake" bucket, and (ii) a strong correlation between the edited region and scene semantics that lets detectors memorize content.

Released splits. Because the pool of source natural images is large, we keep source identities disjoint across the train, validation, and OOD sets so that each fake image corresponds to exactly one natural image and identity bias is avoided. AI Mask Edit and AI Inpainting Edit have distinct semantic labels, and all six groups have equal release-wide counts. The release includes $\mathbf{428{,}580}$ OOD test images, comprising $142{,}860$ held-out Hybrid Edit samples and $285{,}720$ held-out samples associated with SD3, FLUX.1-dev, DeepFillv2, and SD1.5. The remaining $\mathbf{5{,}546{,}808}$ images form the training and validation pool, from which 1% is reserved for validation. Table 5 gives the source composition of the complete release, including the held-out test samples.

Composition.

Table 5: **ForgePan-5.9M source composition.** Natural samples are drawn from OpenImages, OPA, and MS-COCO; face-centric samples are drawn from VGG-Face2 and FaceForensics++. AI MASK EDIT and AI INPAINTING EDIT are distinct supervised semantic classes, produced by mask-based editing and text-conditioned inpainting respectively. HYBRID EDIT is the sixth supervised class and is used for training as well as for evaluation. A dash denotes a source/generator combination the pipeline does not produce. The shaded row is the released manifest.

Source	Generator	NATURAL	PHYS. EDIT	AI EDIT		FULL AI-GEN.	HYBRID EDIT	Row total
				Mask	Inpainting			
OPA Liu et al. (2022a)	SDXL-Turbo	89,000	89,000	89,000	89,000	89,000	89,000	534,000
MS-COCO Lin et al. (2014)	SDXL-Turbo	337,519	337,519	337,519	337,519	337,519	337,519	2,025,114
OpenImages Kuznetsova et al. (2020)	LaMa, Deepfillv2, edgeconnect	42,858	42,858	42,858	–	42,858	42,858	214,290
	SD 3.5	14,286	14,286	14,286	14,286	14,286	14,286	85,716
	SD 3	14,286	14,286	14,286	14,286	14,286	14,286	85,716
	FLUX.1-dev	14,286	14,286	14,286	14,286	14,286	14,286	85,716
	FLUX.1-schnell	14,286	14,286	14,286	14,286	14,286	14,286	85,716
	SD v1.5	14,286	14,286	14,286	14,286	14,286	14,286	85,716
	SD v2.1	14,286	14,286	14,286	14,286	14,286	14,286	85,716
	Kandinsky	14,286	14,286	14,286	14,286	14,286	14,286	85,716
	SDXL-Turbo	–	–	–	42,858	–	–	42,858
OpenImages subtotal		142,860	142,860	142,860	142,860	142,860	142,860	857,160
VGG-Face2 Parkhi et al. (2015)	SDXL-Turbo	213,260	213,260	213,260	426,519	213,260	213,260	1,492,819
FaceForensics++ Rossler et al. (2019)	LaMa, Deepfillv2, edgeconnect	213,259	213,259	213,259	–	213,259	213,259	1,066,295
Released manifest		**995,898**	**995,898**	**995,898**	**995,898**	**995,898**	**995,898**	**5,975,388**

- **Total size:** 5,975,388 images, including 428,580 OOD test images (142,860 held-out HYBRID EDIT images plus 285,720 held-out-generator images from the remaining five classes) and a 5,546,808-image training and validation pool. We reserve 1% of this pool for validation.
- **Supervised classes:** NATURAL, FULLY AI-GENERATED, PHYSICAL EDIT, AI MASK EDIT, AI INPAINTING EDIT, and HYBRID EDIT, the sixth supervised category.
- **Released group counts:** 995,898 images for each of the six groups: NATURAL, PHYSICAL EDIT, AI MASK EDIT, AI INPAINTING EDIT, FULLY AI-GENERATED, and HYBRID EDIT. The two AI-edit classes have separate semantic labels. These are release-wide per-class counts, not training-only counts.
- **Balancing:** The GAN-based inpainters (LaMa, DeepFillv2, and EdgeConnect) cannot be conditioned on text prompts, so they produce no AI INPAINTING EDIT samples. This affects FaceForensics++ and the corresponding OpenImages split. To compensate, we generate additional text-conditioned inpainting samples with SDXL-Turbo: 426,519 on VGG-Face2 (covering both the VGG-Face2 and FaceForensics++ shares) and 42,858 on OpenImages. As a result, every class contains exactly 995,898 images. This keeps the classes balanced and the face/natural-scene proportions consistent across classes, while minimizing generator-specific bias.
- **Ground truth:** For each image we release the pixel-level panoptic label map over the five region classes, the six-way image-level provenance label, the binary edit mask, the generator/editor identifier, the prompt (for AI edits), the masking operator, and an LPIPS quality score.

Collection process. Source images are drawn from multiple public corpora: **OPA** Liu et al. (2022a) (CC-BY 4.0), **MS-COCO** Lin et al. (2014) (Flickr terms), **VGG-Face2** Parkhi et al. (2015) (research-only), **FaceForensics++** Rossler et al. (2019) (research-only), and openimages. Manipulations are generated by our pipeline described in Sec. E.1. We honour opt-out lists published by each source corpus. Table 5 gives the per-class-per-source breakdown.

Preprocessing / cleaning. Every generated sample passes three filters: (i) pixel-level mask-consistency (Hamming-distance $\geq 1\%$ between the source and the edited image restricted to the mask); (ii) perceptual-quality check via LPIPS < 0.28; (iii) pHash blocklist against all evaluation sets.

Uses. The dataset is released for research on image forgery detection and localization, panoptic provenance segmentation, and bias evaluation of forensic classifiers. Commercial use is not permitted.

Distribution. The dataset, panoptic labels, data-generation pipeline, pretrained weights and evaluation scripts are distributed under a research-only license on the project page. A persistent DOI and a DataCite record will be provided at camera-ready.

Algorithm 1 GENERATEFORGEPANSAMPLE: generation of an image, its image-level six-class provenance label, and its pixel-level panoptic annotation in ForgePan-5.9M. Hybrid samples contain surviving physical *and* AI edits; their segments keep the operator classes PHYSICAL EDIT, AI MASK EDIT and AI INPAINTING EDIT in L, and only the image-level label y_{img} is set to HYBRID EDIT (class 5).

Require: Source image x; base provenance $b \in \{0, 1\}$ (NATURAL or FULLY AI-GENERATED); configured full-image generator G; ordered edit schedule $\mathcal{E} = ((m_j, c_j, \text{op}_j))_{j=1}^{J}$, where $c_j \in \{2, 3, 4\}$.
Ensure: Image $\tilde{x}$; image-level label y_{img}; panoptic annotation (L, S); metadata η, or REJECT.
Masks are sampled once and reused across aligned branches. Accepted samples have no surviving edits, one surviving operator class, or both physical and AI operator classes.

$\tilde{x} \leftarrow x$
$\mathcal{T} \leftarrow [\,]$
$\eta \leftarrow \{\text{base provenance} : b,\ \text{turn records} : [\,]\}$
if $b = 1$ **then**
  $\tilde{x} \leftarrow \text{GENERATEFULL}(G, x)$
  Record generator configuration and conditioning in η
end if
for $j = 1, \ldots, J$ **do**
  if $c_j = 2$ **then**
    $\tilde{x} \leftarrow \text{PHYSICALEDIT}(\tilde{x}, m_j, \text{op}_j)$
  else if $c_j = 3$ **then**
    $\tilde{x} \leftarrow \text{MASKEDIT}(\tilde{x}, m_j, \text{op}_j)$
  else if $c_j = 4$ **then**
    $\pi_j \leftarrow \text{IDENTITYPROMPTFROM}(\tilde{x}, m_j)$
    $\tilde{x} \leftarrow \text{INPAINT}(\tilde{x}, m_j, \pi_j, \text{op}_j)$
  end if
  Append (m_j, c_j) to $\mathcal{T}$
  Append turn index, mask, original operator class c_j, operator, generator, and conditioning to η
end for
if not PASSQUALITYCHECKS$(\tilde{x}, x, \mathcal{E})$ **then**
  return REJECT
end if
$(L, S, \mathcal{C}_{\text{vis}}, \rho, \text{hyb}) \leftarrow \text{BUILDPANOPTIC}(b, \mathcal{T})$
$\mathcal{C}_{\text{vis}}$ contains the operator classes surviving overlap resolution and void filtering, which L retains unchanged; ρ records each surviving thing segment's originating turn and operator class; hyb *flags the co-occurrence of a physical and a generative operator.*
if $\mathcal{C}_{\text{vis}} = \varnothing$ **then**
  $y_{\text{img}} \leftarrow b$
else if $2 \in \mathcal{C}_{\text{vis}}$ **and** $\mathcal{C}_{\text{vis}} \cap \{3, 4\} \neq \varnothing$ **then**
  $y_{\text{img}} \leftarrow 5$ ▷ Image level only; L keeps operator classes
else if $|\mathcal{C}_{\text{vis}}| = 1$ **then**
  $y_{\text{img}} \leftarrow$ the unique element of $\mathcal{C}_{\text{vis}}$
else
  return REJECT ▷ AI-only mixture $\{3, 4\}$ is outside this taxonomy
end if
Store ρ and $\mathcal{C}_{\text{vis}}$ in η
return $(\tilde{x}, y_{\text{img}}, (L, S), \eta)$

Maintenance. We commit to maintaining ForgePan-5.9M with yearly updates that include new diffusion generators as they become available and to tracking opt-out requests from faces in the VGG-Face2 source.

E.1 DATA GENERATION PIPELINE DETAILS

Algorithm 1 gives the pseudocode of one sample of the data-generation pipeline whose visual overview is in Fig. 3, and Algorithm 2 gives the routine that converts the resulting per-turn edit record into the panoptic annotation (L, S) of Sec. D. We give the second routine explicitly because it is what makes the annotation panoptic rather than semantic: it is where segment identifiers are assigned, where the thing/stuff distinction is applied, and where several regions of the *same* class become several distinct segments.

Algorithm 2 BUILDPANOPTIC: construction of the pixel-level panoptic annotation. Later turns take precedence. Each visible edited component receives a distinct segment ID and *retains its operator-specific class* $c_j \in \{2, 3, 4\}$, whether or not physical and AI edits co-occur in the same image. Unedited pixels retain base provenance b. The routine only *reports* co-occurrence, via $\mathcal{C}_{\text{vis}}$ and the flag hyb that the caller turns into the image-level HYBRID EDIT verdict; class 5 is never written into L. Segment provenance is additionally logged in ρ.

Require: Base provenance $b \in \{0, 1\}$; ordered edit record $\mathcal{T} = ((m_j, c_j))_{j=1}^{J}$, where $m_j \subseteq \Omega$ and $c_j \in \{2, 3, 4\}$; pixel lattice Ω.
Ensure: Class map L over classes $\{0, \dots, 4\}$; segment-ID map S; surviving operator classes $\mathcal{C}_{\text{vis}}$; segment provenance metadata ρ; hybrid co-occurrence flag hyb.

$L \leftarrow \varnothing$ everywhere; $S \leftarrow 0$ everywhere; next $\leftarrow 1$; $\rho \leftarrow \{\}$
$\varnothing$ denotes the void class; segment ID 0 is reserved for void pixels.
▷ (i) Resolve overlap and assign provisional classes
for $j = 1, \dots, J$ **do**
  $v_j \leftarrow m_j \setminus \bigcup_{k=j+1}^{J} m_k$
  for all $r \in \mathrm{CC}_8(v_j)$ **do**
    if $|r| \geq 128$ **then**
      $L[r] \leftarrow c_j$
      $S[r] \leftarrow$ next
      $\rho[\text{next}] \leftarrow (j, c_j)$ ▷ Originating turn and original class
      next $\leftarrow$ next $+ 1$
    end if ▷ Smaller components remain void
  end for
end for
▷ (ii) Unedited base forms one stuff segment
$u \leftarrow \Omega \setminus \bigcup_{j=1}^{J} m_j$
if $u \neq \varnothing$ **then**
  $L[u] \leftarrow b$
  $S[u] \leftarrow$ next
end if
▷ (iii) Exclude boundary ambiguity
$(L_0, S_0) \leftarrow (L, S)$
$\Gamma \leftarrow$ INTERNALBOUNDARIES(L_0, S_0)
$B \leftarrow \{p \in \Omega : \mathrm{dist}(p, \Gamma) \leq 2 \text{ pixels}\}$
$L[B] \leftarrow \varnothing$; $S[B] \leftarrow 0$
▷ (iv) Determine surviving original provenance
$\mathcal{C}_{\text{vis}} \leftarrow \{c \in \{2, 3, 4\} : \exists p \in \Omega,\ L(p) = c\}$
Remove entries $\rho[s]$ for which $\{p \in \Omega : S(p) = s\} = \varnothing$
▷ (v) Report hybrid co-occurrence; do not relabel
hyb $\leftarrow \big(2 \in \mathcal{C}_{\text{vis}}$ **and** $\mathcal{C}_{\text{vis}} \cap \{3, 4\} \neq \varnothing\big)$
*L is deliberately **not** modified here. Every surviving edited segment keeps the operator class assigned in step (i), so a hybrid image is annotated with* PHYSICAL EDIT *and* AI MASK/INPAINTING EDIT *segments side by side, each with its own identifier in S. Collapsing them onto one class would reintroduce exactly the undifferentiated "fake" mask that Sec. 1 argues against, and would leave the co-occurring localization of Table 4 unscorable.* HYBRID EDIT *(class 5) is an image-level verdict, emitted by the caller from* hyb.
return $(L, S, \mathcal{C}_{\text{vis}}, \rho, \text{hyb})$

Worked example. Consider a natural image subjected to three editing turns: two physical splices ($c = 2$), followed by one diffusion inpainting ($c = 4$) that partially overlaps the second splice. Later turns take precedence in overlapping pixels. Assume that each edited region remains connected and retains non-void pixels after the size and boundary filters. The surviving original operator classes are therefore $\mathcal{C}_{\text{vis}} = \{2, 4\}$, and the image-level label is $y_{\text{img}} = 5$ (HYBRID EDIT).

The panoptic annotation contains three distinct thing segments that *keep their operator classes* – two PHYSICAL EDIT segments ($L = 2$) and one AI INPAINTING EDIT segment ($L = 4$) – together with one NATURAL stuff segment ($L = 0$) covering the non-void unedited remainder. The three edited segments also carry distinct identifiers in S, so the two splices are scored as two segments rather than one. Class 5 never appears in L: the co-occurrence of 2 and 4 in $\mathcal{C}_{\text{vis}}$ sets the *image-level* label $y_{\text{img}} = 5$ and nothing else. This is precisely what the localization claim of Sec. 1 asserts and what Table 4 scores: a single forward pass has to recover the physical and the AI region of one image *as*

different classes, so the operator distinction is a graded prediction rather than a convention folded into the ground truth. Metadata ρ additionally records each segment's originating turn index $(1, 2, 3)$ and operator.

Released annotation format. Each sample ships a two-channel PNG carrying L in the first channel and S in the second, with 255 reserved for void, alongside a JSON record listing every segment as a tuple (identifier, class, area in pixels, originating turn index, operator, generator). The per-turn masks $\{m_j\}$ and the BUILDPANOPTIC implementation are released with the dataset, so the segment decomposition behind every reported PQ value can be regenerated exactly rather than inferred from the class map. Some samples from the dataset, the edited region strengths and ground truth masks for both hybrid and shared masks are shown in Fig. 6- 8.

The *identity-preserving prompt* used for the $c{=}4$ branch is computed by running a CLIP-based open-vocabulary detector over x, filtering the detections to those intersecting the sampled mask m, and joining their category names into the positive prompt. A negative prompt ("low quality, distorted, watermark, jpeg artefacts") is also passed to the inpainter. This keeps the semantic content under m close to the original and forces the model to learn manipulation fingerprints rather than object semantics.

Mask family $\mathcal{M}$. Masks are sampled with equal probability from elliptic (axis ratios $\in [0.3, 3]$), Voronoi patches, free-form brush strokes as in Liu et al. (2018), object-aligned masks from SAM Kirillov et al. (2023) restricted to OPA donors, face-aligned masks from 68-landmark convex hull, and axis-aligned rectangles. Mask area is uniformly sampled on $\log$-scale between 0.5% and 90% of the image.

Face-centric share. $426{,}519$ of the $995{,}898$ images in each class (42.8%) are face-centric. For face sources (VGG-Face2 / FaceForensics++), the mask sampler is biased towards face-aligned masks for $c \in \{2, 3, 4\}$. This ensures the dataset reflects the empirical distribution of face-forgery evidence.

F DETAILS ABOUT THE IEDAL2 CHALLENGE

IEDAL2 evaluates submitted detectors on private, organizer-held data through the UL DSRI Dyff platform. Participants submit a runnable Docker container implementing a standardized HTTP/JSON interface; no training dataset is supplied, and the evaluation images are not released. The challenge distinguishes five image categories: Natural, FullySynthesized, LocallyEdited (traditional manipulation), LocallySynthesized, and LocallyInstructSynthesized (instruction-guided synthesis), with each image containing manipulations from at most one category. Detection is binary—natural versus all manipulated categories—whereas image-level classification is five-way. Localization evaluates manipulated pixels as a single foreground category, rather than predicting manipulation-specific pixel labels. The published evaluation lists detection accuracy, balanced accuracy, and AUC, together with localization IoU and pixel-level F1; conventionally, balanced accuracy is the mean of sensitivity and specificity, while pixel F1 and IoU are (2TP/(2TP+FP+FN)) and (TP/(TP+FP+FN)), respectively. Thus, our reported 87.7% balanced accuracy and 60.3% localization F1 support transfer to independently constructed manipulation data but do not independently validate hybrid-edit separation or multiclass panoptic attribution, which are evaluated on ForgePan-5.9M. The challenge maintains separate capability rankings, and its rules designate the private leaderboard as the basis for final ranking. The challenge includes samples generated using commercial generators such as Nano Banana Pro Google DeepMind (2025), Luma Uni Luma AI (2026), Seedream 4.5 ByteDance Seed (2025), and Grok Imagine xAI (2026b), as well as open-weight generators such as Hidream O1 HiDream.ai (2026), Wan 2.7 Image Pro Alibaba Cloud (2026), Hunyuan Image V3 Tencent (2025), and Qwen Image Edit 2511 Qwen Team (2025). The evaluation of the proposed approach on the IEDAL2 challenge further specifies the effectiveness of the proposed approach in generalization to unseen generators. These details were disclosed after the challenge was closed and the final submission was made. These were not used for training, model selection, or threshold tuning.

G Additional Ablations

All sweeps in this section report mPQ* on the full ForgePan-5.9M-test set under the protocol of Table 3, so every entry is directly comparable to the full model's 90.75 mPQ* (equivalently 91.39 mPQ_{5c}). The endpoints that coincide with a row of Table 3 reproduce that row exactly: λ_c=0 in Table 6 is the *w/o SupCon* variant (88.10). The full-model setting of each sweep (λ_c=0.5, Q=32, and the 924k-per-class pool) reproduces the 90.75 mPQ* of Table 3; note that it is the rightmost column only in Table 7. Two sweep endpoints exceed the full model (λ_c=1.0 at 91.52 and Q=64 at 91.62); we keep λ_c=0.5 and Q=32 as the reported configuration because they were fixed before the test evaluation, and we mark the full-model column rather than the sweep maximum.

Table 6: Effect of λ_c (SupCon loss weight) on mPQ*. λ_c=0 is the *w/o patch-wise SupCon* row of Table 3; λ_c=0.5 is the full model.

λ_c	0.0	0.1	0.25	0.5	1.0	2.0
mPQ*	88.10	89.70	90.18	**90.75**	91.52	90.56

Table 7: Effect of training corpus size on mPQ*. We downsample ForgePan-5.9M uniformly per class; the rightmost column is the full training pool (5,546,808 images, ≈ 924k per class on average) and reproduces the full model of Table 3.

Samples per class	10k	50k	100k	250k	500k	924k
mPQ*	66.85	77.10	82.55	85.96	89.06	**90.75**

Table 8: Effect of the number of object queries Q. Q=32 is the setting used throughout the paper and reproduces the full model of Table 3.

Q	4	8	16	32	64	128
mPQ*	83.83	86.82	90.02	**90.75**	91.62	91.20

Loss component ablation. The per-term ablation of Eq. 4 is reported in the main paper (Table 3, **Obj.** block), which toggles $\mathcal{L}_{\text{supcon}}$, $\mathcal{L}_{\text{boundary}}$, $\mathcal{L}_{\text{DC}}$, $\mathcal{L}_{\text{cls}}$ and $\mathcal{L}_{dice}$ one at a time and additionally reports the CE-only mask-supervision variant, in which queries are assigned by Hungarian matching on the mask cost alone and the image-level class is read off the $\mathcal{L}_{\text{cls}}$ head at inference. We do not duplicate those numbers here.

H Failure-Mode Analysis

We audit 1,000 errors on the validation split and cluster them into five modes.

1. **Tiny edits ($r < 1\%$)** account for 37% of errors. The $F_{1/8}$ level is down-sampled, so edits smaller than ~ 30 pixels escape the decoder's attention footprint.
2. **Global colour grades** account for 18% of errors, mostly false-positive on PHYSICAL. Our mask operator for colour-grading covers the entire image, so the model occasionally flags an aggressive natural grade as a physical edit.
3. **AI-generated regions on cartoon / painting input** account for 11% of errors. The DINOv3 encoder was pretrained on natural images and has weaker features for synthetic styles.
4. **Very large AI edits ($r > 70\%$)** account for 9% of errors, confused with FULL. This is the most intuitive confusion.
5. **Mask-border ambiguity** accounts for the remaining 25% of errors – correct class, pixel boundary off by 3–8 pixels.

These failure modes motivate two extensions: a higher-resolution pyramid level for tiny edits, and an explicit global-consistency head to disentangle colour-grading from splicing.

I HYPERPARAMETERS

Table 9: Hyperparameters used to reproduce ForgeFormer.

Optimizer	AdamW (β_1=0.9, β_2=0.999, wd = 0.05)
LR (decoder / heads)	1×10^{-4}
LR (LoRA on encoder)	2×10^{-5}
Schedule	cosine, 1500-step warmup
Batch size	64
Image resolution	512×512
Epochs	24
Mixed precision	bf16
Gradient clip	1.0
LoRA rank	16
Decoder layers N	9
Object queries Q	32
Hidden dim d_m	256
Encoder	DINOv3-ViT-H+ encoder with dimension 1280
Pyramid levels	$\{1/8, 1/16, 1/32\}$
Loss weights	λ_{ce}=2.0, λ_{dice}=5.0, λ_b=1.0, λ_c=0.5, λ_{cls}=2.0
SupCon τ	0.07
Classification label smoothing	0.15
Total non-trainable arameters	840.6M
Trainable parameters (LoRA + decoder + heads)	5.24M
Inference memory (fp16 / int8)	6.6 GB / 4.9 GB
Hardware	8 × NVIDIA RTX PRO 6000 Blackwell
Walltime	$\approx$ 4 days

J SCOPE OF THE ALIGNMENT CLAIM AND RESIDUAL CONFOUNDS

We describe ForgePan-5.9M as *operator-aligned* rather than bias-free. This section states what that control buys and what it leaves open.

What alignment controls. Within an editing turn, the same mask geometry m is replayed across the operator branches, and only the operator that regenerates the masked pixels changes. This removes the dominant shortcut in prior localization benchmarks, where a small gallery of objects is repeatedly pasted so that a detector can learn *what* tends to be edited in place of *that* something was edited. It is a control on the relationship between the mask and the content placed inside it, and nothing more.

Table 10: Cross-benchmark pixel-F1. Each baseline is trained on its *own* training set or with ForgePan-5.9M added (**+FP**). Superscripts give the absolute gain. ForgeFormer is trained only on ForgePan-5.9M. **Bold** and underline mark the best and second-best results.

Method	**Train**	**CASIA v2**	**Cover-age**	**IMD 2020**	**DiffSeg 30K**
MVSS-Net	own	52.1	41.8	47.9	38.2
	+FP	61.9	49.7	55.4	55.1
TruFor	own	68.4	52.6	58.2	46.8
	+FP	74.5	58.8	63.1	62.0
SIDA	own	72.8	58.9	61.9	55.7
	+FP	<u>76.9</u>	<u>63.4</u>	<u>65.8</u>	<u>68.2</u>
ForgeFormer	FP	**79.4**	**64.1**	**68.5**	**73.3**

What it does not control.

1. *Mask geometry and source semantics remain dependent.* The query q is sampled from the categories the source image already annotates, and the segmentation that produces m is conditioned on that image. Drawing q before the manipulation removes the dependence on the *inserted* content, not the dependence on the *source* scene, so mask geometry still correlates with scene composition.
2. *Source composition is not matched across classes.* As Table 5 shows, AI INPAINTING EDIT draws $426{,}519$ samples from VGG-Face2 and none from FaceForensics++, whereas every other class draws $213{,}260$ from VGG-Face2 and $213{,}259$ from FaceForensics++. The face-versus-scene proportion is matched, but the two face corpora differ in capture conditions, compression history and identity distribution, so part of what separates AI INPAINTING EDIT from the other classes may be source-corpus statistics rather than an inpainting fingerprint. The same imbalance means the one-to-one correspondence between a natural image and its manipulated counterparts, which holds elsewhere in the release, does not hold for this class: its VGG-Face2 share is roughly two inpainted samples per source image.
3. *Generator availability is coupled to class and to source.* The GAN inpainters cannot be conditioned on text and therefore contribute no AI INPAINTING EDIT samples at all, and the compensating samples are produced entirely with SDXL-Turbo. Generator identity is consequently predictable in part from class and source, a confound that mask-level alignment does not touch.
4. *Branch-specific post-processing is unmeasured.* The physical-edit branch ends in a JPEG recompression at $Q \sim \mathcal{U}(75, 95)$ that the generative branches do not receive, and the branches differ in how they composite and blend at the mask boundary. Either could act as a per-class cue. We have not tested whether a detector restricted to compression statistics or to boundary structure alone can separate the branches, and until that is done, the claim that the model must be reading an intrinsic manipulation fingerprint is an assumption rather than a result.

Consequences for the claims in this paper. We therefore do not claim that bias removal is the primary driver of generalization. The experiment that would support such a claim is a controlled comparison between an aligned and an unaligned corpus matched on size, source composition, generator inventory, and post-processing, with alignment as the only difference. We do not run it, and the cross-benchmark gains of Table 10 are consistent with an alignment effect without isolating one.

A second open comparison. As noted in Sec. 3, the binary baselines are adapted to a class-aware panoptic target by a conversion we impose, so their margins confound representation quality with output-space expressiveness. We regard this and the aligned-versus-unaligned experiment above as the two most important additions for a future version.

K LICENSING

Our data generation code, pretrained weights, and evaluation scripts are released under Apache-2.0. The ForgePan-5.9M image corpus is released under a research-only license that forbids commercial use and redistribution with identity labels beyond what the underlying source corpora allow. OPA Liu et al. (2022a) inherits its CC-BY 4.0 source. MS-COCO source images follow Flickr terms. VGG-Face2 and FaceForensics++ images follow their respective research-only licenses; derived manipulations maintain the parent licenses.

K.1 BROADER IMPACT

A robust, panoptic forgery detector has obvious defensive benefits: improved image provenance for journalism, more precise moderation decisions, and better forensic chain of custody. It also carries dual-use risks: any detector provides, in principle, an adversarial gradient for evasion. We partially mitigate this by (i) releasing only model weights, not white-box training checkpoints tied to specific moderation products; (ii) documenting failure modes openly; (iii) restricting the dataset license to non-commercial research; and (iv) encouraging the community to report evasion attempts back to the benchmark maintainers so that successive releases keep pace.

L ROBUSTNESS TO POST-PROCESSING

Table 11 reports the per-perturbation mPQ* drop on the full ForgePan-5.9M-test set, using the same protocol and the same clean-model operating point as Table 3. We apply six standard post-processing

perturbations at a single fixed strength each—JPEG compression at quality 50, additive Gaussian noise with $\sigma{=}5/255$, a $5{\times}5$ Gaussian blur, half-resolution bilinear resize, WebP compression at quality 75, and a random $0.8\times$ center crop followed by upscaling to the original resolution—and report the mPQ drop relative to the clean baseline. ForgeFormer loses on average only 3.5 mPQ* points across the six perturbations, compared with 6.7–16.0 for the binary-only baselines.

Table 11: Robustness on the full ForgePan-5.9M-test set: mPQ* under six post-processing perturbations, using the composite summary defined in Table 4. Δ is the absolute mPQ* drop from each method's clean-image baseline. Lower-magnitude drops are better.

Method	Clean mPQ*	JPEG-50	Noise $\sigma{=}5$	Blur $5{\times}5$	Resize $0.5\times$	WebP-75	Crop $0.8\times$	Avg. Δ
MVSS-Net	39.01	-19.4	-15.2	-14.1	-17.8	-14.6	-14.9	-16.0
TruFor	45.03	-11.7	-8.5	-7.2	-9.8	-6.9	-5.1	-8.2
SIDA	50.86	-8.4	-7.1	-5.8	-7.6	-6.0	-5.3	-6.7
ForgeFormer (ours)	**90.75**	$\mathbf{-4.6}$	$\mathbf{-3.8}$	$\mathbf{-3.1}$	$\mathbf{-4.0}$	$\mathbf{-2.9}$	$\mathbf{-2.6}$	$\mathbf{-3.5}$

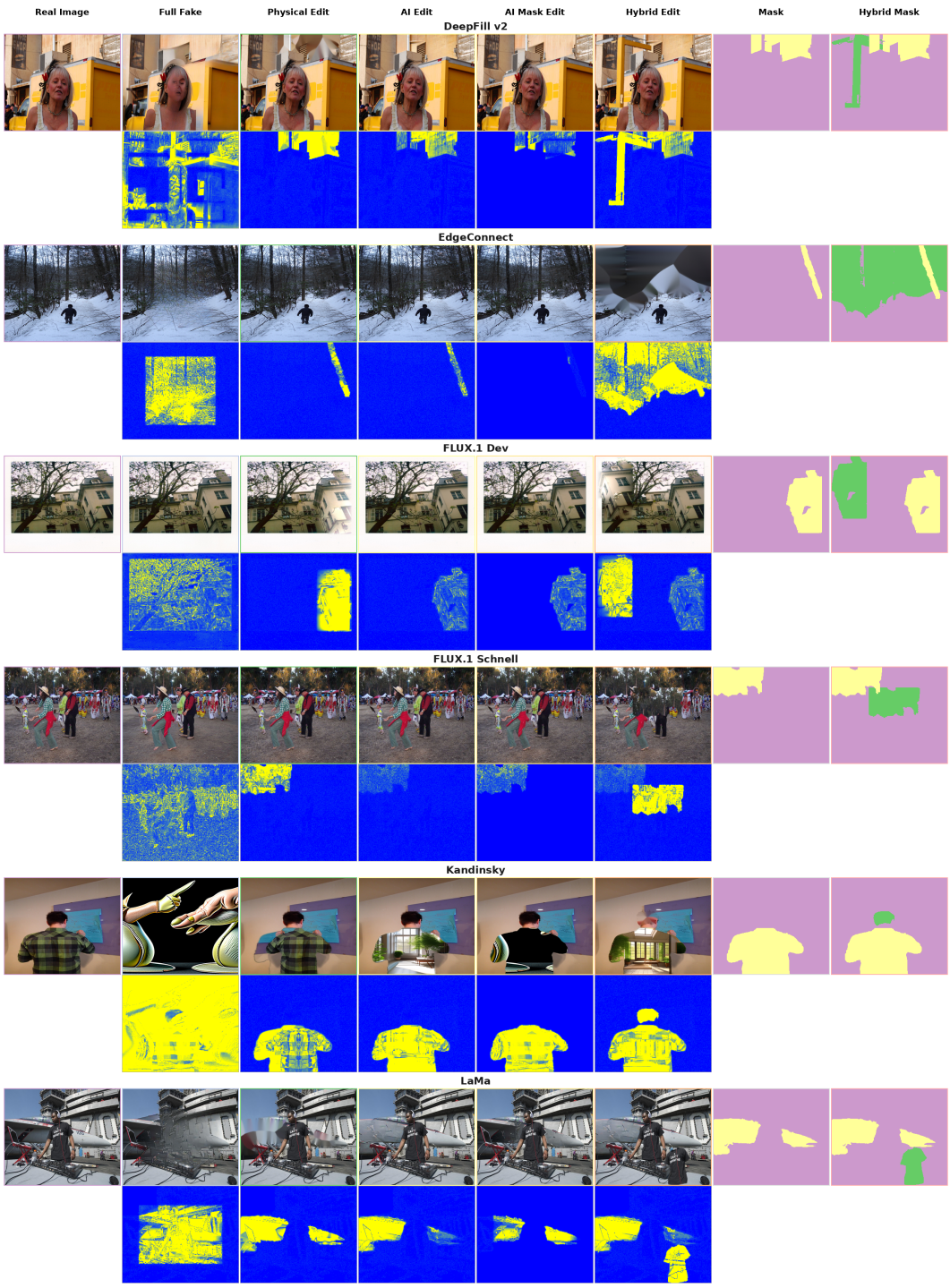


Figure 6: **Visualization of ForgePan-5.9M data samples.**

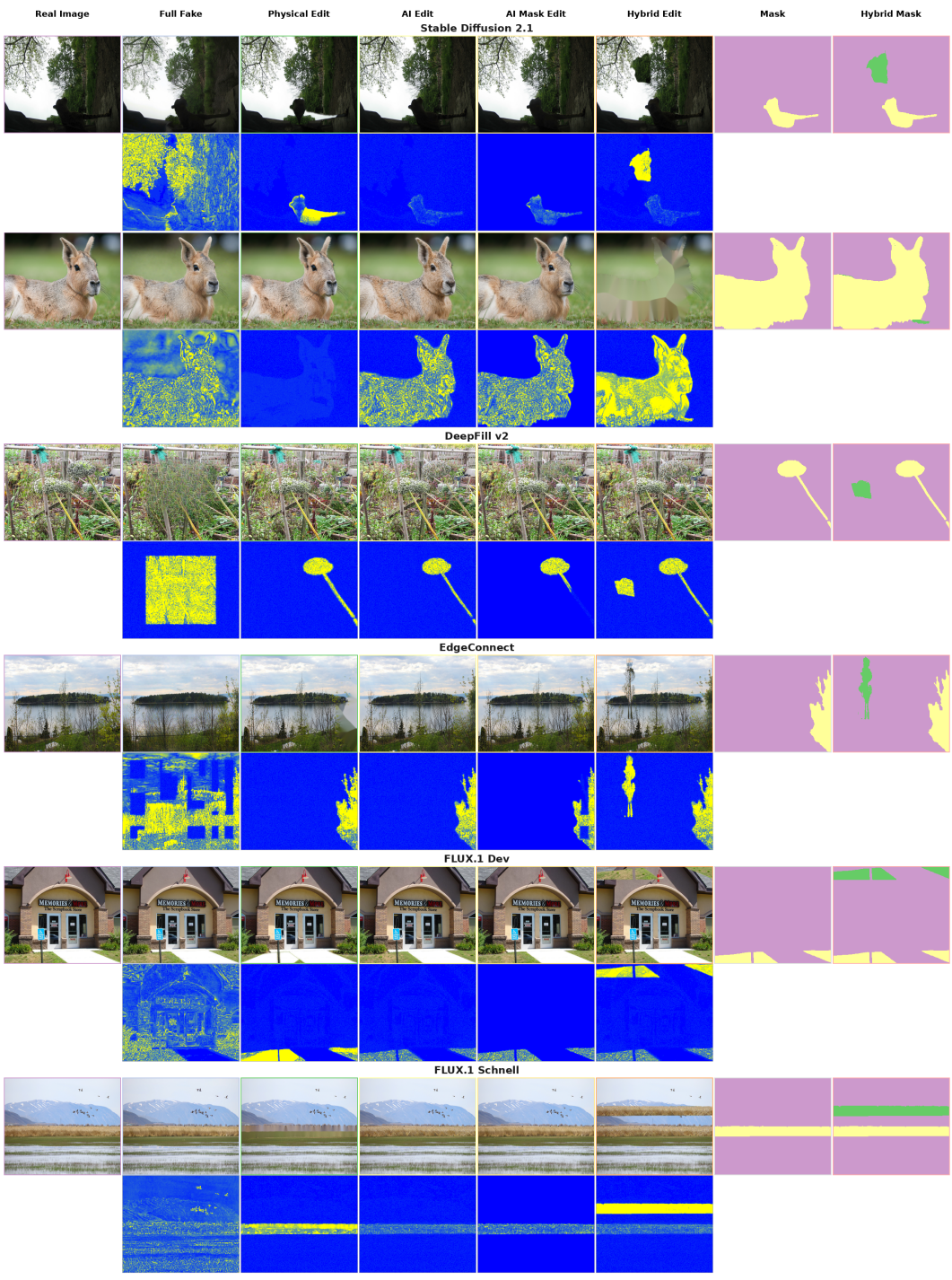


Figure 7: **Visualization of ForgePan-5.9M data samples.**

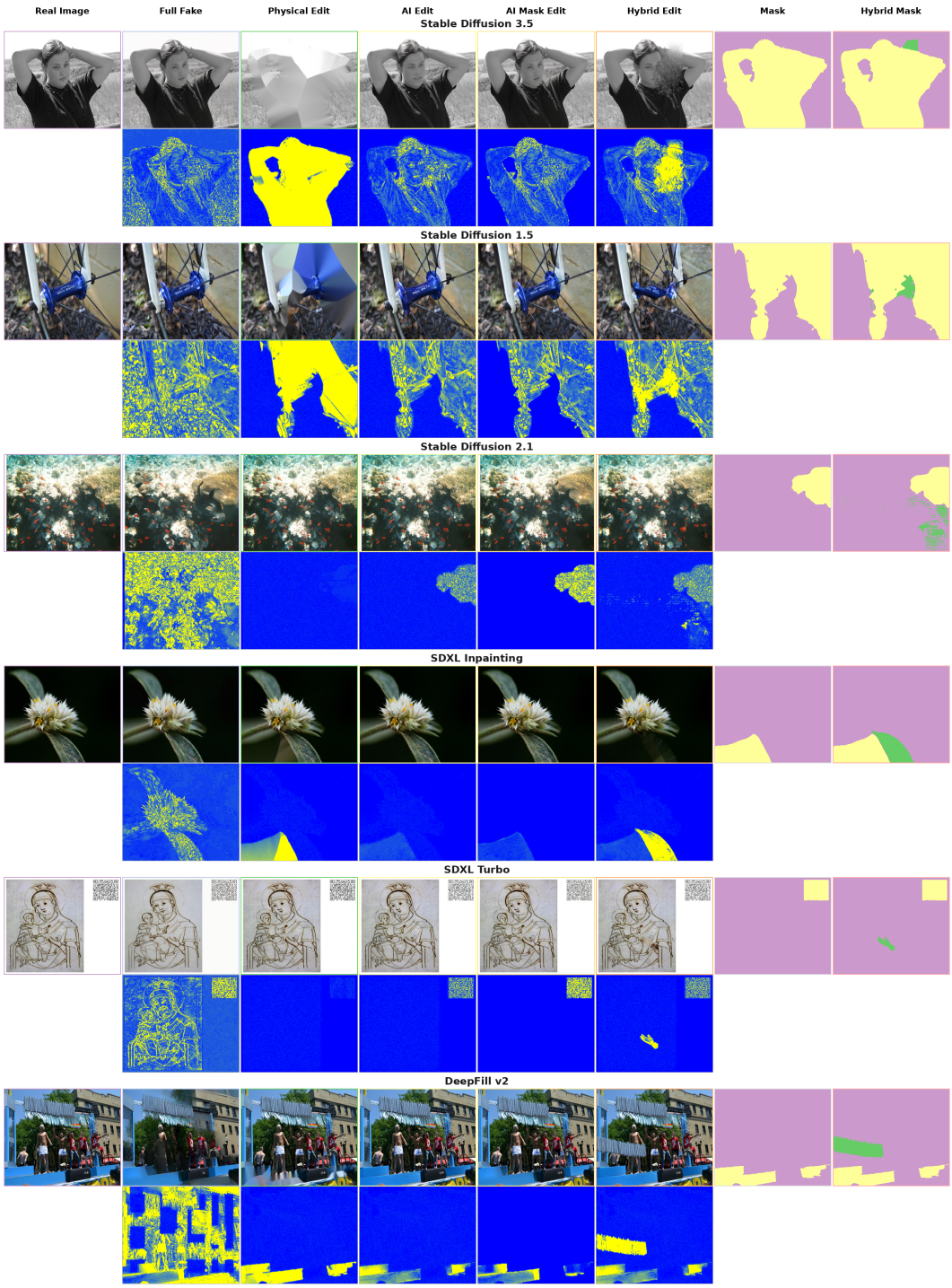


Figure 8: **Visualization of ForgePan-5.9M data samples.**