DINOv3 Beats Specialized Detectors: Revisiting Image Manipulation Localization from a Foundation Model Perspective

Anonymous authors
Paper under double-blind review

Abstract

Image manipulation localization (IML) identifies regions that have been spliced, copied, removed, or otherwise edited. Its labels describe provenance rather than semantic identity, so existing systems rely on specialized forensic cues. Frozen DINOv3 features, despite strong readouts on general vision tasks, localize manipulations poorly. We ask whether a general backbone can make editing evidence readable without a dedicated forensic front end. A stronger decoder helps only in part: a global Transformer head improves frozen features, but adapting the encoder yields a far larger gain. We call this contrast the forensic readability gap: frozen features contain editing information, but it is difficult to read with one linear rule shared across images. We therefore propose **DINOv3-IML**, a simple QKV-LoRA adaptation with a shallow convolutional head and no forensic feature extractor. Adaptation makes editing evidence readable through a shared linear rule and aligns editing directions across images; once the encoder is adapted, a global head brings no further gain. DINOv3-IML reaches 0.85 and 0.77 pixel F1 under the CAT-Net and MVSS protocols, surpassing the best prior methods by 12 and 24 points, respectively, and establishes a simple yet strong baseline for IML.

1 Introduction

Images can now be generated, edited, and recombined with unprecedented realism. As visual media enters news, social platforms, scientific communication, and digital evidence, determining whether an image has been altered is essential for trustworthy multimedia (Verdoliva, 2020). Image manipulation localization (IML) goes beyond an image-level verdict by identifying the pixels or regions that were inserted, removed, copied, or otherwise modified.

IML is commonly formulated as segmentation, but its labels describe provenance rather than semantic identity. The same local appearance can be authentic or manipulated, while regions with different content can share an editing history. Evidence may lie in inconsistencies in sensor or compression statistics, camera characteristics, boundaries, or relationships among regions (Hsu & Chang, 2006; Cozzolino & Verdoliva, 2018; Verdoliva, 2020). Accordingly, IML systems have long emphasized low-level forensic cues, using residual or frequency-sensitive features, multi-view branches, and learned noise fingerprints to support localization (Kwon et al., 2022; Dong et al., 2023; Guillaro et al., 2023; Ma et al., 2024a). Yet many ultimately train a visual backbone and pixelwise prediction head in a formulation familiar from semantic segmentation. This raises a representation question: can a backbone make editing relationships readable without a specialized forensic front end?

DINOv2 and DINOv3 make this question timely: their frozen self-supervised features support strong readouts on general vision tasks (Oquab et al., 2024; Siméoni et al., 2025). The same approach is much less effective for IML. Under the IMDL-BenCo CAT-Net protocol (Ma et al., 2024b), a three-layer convolutional head on frozen DINOv3 features reaches 0.44 pixel F1, behind the 0.536 reported for the ResNet-based MVSS-Net (Tables 1 and 6). One possibility is that the requisite editing information is present but requires a more capable decoder to compare regions. A matched ViT-L experiment tests this possibility: replacing a convolutional head with a two-layer global Transformer raises frozen-feature F1 from 0.32 to 0.47. Adapting the encoder with low-rank updates, however, raises the same Transformer configuration to 0.76. Encoder adaptation yields the larger gain.

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.

We call this contrast the *forensic readability gap*. To study it independently of the mask decoder, we define cross-image linear readability as the best mean per-image AP achievable by a single linear rule shared across images, estimated with a shared probe. The underlying criterion is invariant to invertible global affine reparameterizations of the features. Frozen features yield about 0.50 AP, above the 0.14 AP of a constant predictor; after adaptation, AP rises to 0.92. MLP and convolutional probes show the same broad contrast. Editing information is thus present in frozen features, but adaptation makes it substantially easier to read through a consistent rule across images.

We therefore introduce **DINOv3-IML**, a simple yet effective model that adapts DINOv3's query, key, and value projections with low-rank updates and predicts the mask with a shallow convolutional head. It requires no dedicated forensic feature extractor. Under the CAT-Net protocol, it reaches 0.847 pixel F1, 12.2 points above the strongest prior method; a variant with a linear mask head already reaches 0.8145. Under the MVSS protocol, which trains on CASIAv2 alone, it reaches 0.774 pixel F1, 24.4 points above the strongest prior method, and leads on all five test sets. The linear-head result indicates that the adapted encoder can support localization without a complex mask decoder.

Further analyses show how adaptation improves this readout. The cross-image probe gain holds on all four CAT-Net test sets and persists after feature standardization or whitening. The mean alignment between each image's edit–background direction and the source direction rises from 0.21 to 0.78. Both Q/K and V updates contribute to the gain, while a few shared directions of the adaptation residual suffice for strong discrimination. In matched comparisons, a global Transformer head scores 0.76 after adaptation, versus 0.79 for the shallow convolutional head. LoRA also outperforms full fine-tuning with a lower backbone learning rate (0.79 versus 0.67) while using about 1/34 as many trainable parameters.

These findings point to a concrete challenge for future IML models: how to control the roles of semantic context and cross-region evidence in deciding editing provenance. Semantic similarity can guide which regions to compare, but cannot establish whether they share an editing history. Our results suggest that targeted encoder adaptation can shape these comparisons and the information exchanged between regions, while a lightweight head refines the spatial mask.

Our contributions are:

- We identify a *forensic readability gap* in frozen foundation-model features and quantify it with a shared linear rule across images. Decoder and probe comparisons show that frozen features contain editing information, but make it difficult to recover consistently. This supports studying IML as relational provenance inference, beyond the detection of isolated low-level cues.
- We introduce DINOv3-IML, a simple QKV-LoRA adaptation with a shallow mask head and no dedicated forensic feature extractor. It exceeds the strongest prior methods by 12.2 and 24.4 pixel-F1 points under the CAT-Net and MVSS protocols, respectively; even its linear-head variant surpasses the strongest prior CAT-Net result.
- We show that adaptation improves cross-image readout and aligns edit–background directions. Analyses of Q/K and V updates, residual directions, and decoder capacity connect this change to a practical design principle: shape semantic and relational evidence within the encoder, then use a lightweight head for spatial refinement.

2 Related Work

Forensic cues and localization architectures. Image-forensic models exploit complementary evidence about manipulation. Constrained convolutions suppress image content and learn prediction-error filters (Bayar & Stamm, 2016). Noiseprint learns camera-model fingerprints (Cozzolino & Verdoliva, 2018), and CAT-Net uses JPEG compression artifacts (Kwon et al., 2022). Learned self-consistency uses EXIF supervision to compare imaging pipelines across patches (Huh et al., 2018), while SPAN models patch relationships through a pyramid of local self-attention (Hu et al., 2020). MVSS-Net combines multiple views and scales (Dong et al., 2023); PSCC-Net models spatial-channel correlations (Liu et al., 2022); TruFor incorporates reliability estimation (Guillaro et al., 2023). Transformer-based approaches include IML-ViT (Ma et al., 2024a), Mesorch (Zhu et al., 2024), and M2SFormer (Nam et al., 2025). IMDL-BenCo supplies common implementations

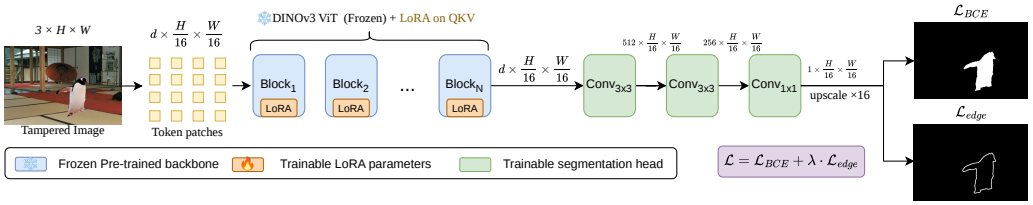

Figure 1: DINOv3-IML adapts a frozen DINOv3 encoder with QKV-LoRA and reads its final patch features with a shallow head. Pixel and boundary-weighted BCE supervise the same logits.

and evaluation protocols (Ma et al., 2024b). Our study examines how task adaptation makes editing status readable from foundation features with a shallow prediction head.

Dense self-supervised representations. DINO demonstrated spatial structure in self-supervised ViT features (Caron et al., 2021). iBOT learns patch-level features by matching teacher predictions at masked positions (Zhou et al., 2022); MAE instead reconstructs missing pixels from visible patches (He et al., 2022). These objectives provide different forms of local supervision without manual labels. DINOv2 developed general representations for image-level and dense prediction (Oquab et al., 2024). DINOv3 further develops dense representations at scale (Siméoni et al., 2025). DinoLizer applies LoRA to DINOv2 and uses a linear patch head for generative inpainting localization (Doi et al., 2026). We study general manipulation localization under the CAT-Net and MVSS protocols and ask whether task adaptation makes editing evidence readable across images.

Parameter-efficient adaptation. Visual Prompt Tuning learns additional input tokens while keeping the vision backbone frozen (Jia et al., 2022). LoRA trains low-rank weight updates while retaining the original parameter matrices (Hu et al., 2021). DoRA separates weight magnitude and direction during adaptation (Liu et al., 2024), whereas AdaptFormer adds parallel adaptation modules to vision transformers (Chen et al., 2022). We use LoRA on DINOv3's QKV projections and compare frozen-backbone and full-fine-tuning configurations. Additional DoRA and AdaptFormer-style runs in Appendix A.5 extend this comparison under the same training recipe.

3 DINOv3-IML and Forensic Readability Analysis

Given an RGB image $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$, we predict a manipulation probability $\hat{\mathbf{M}} \in [0,1]^{H \times W}$. Our approach has two parts. The first is DINOv3-IML (Figure 1), a deliberately simple localizer that combines a dense representation from a vision foundation model (Section 3.1), constrained low-rank task adaptation (Section 3.2), and a shallow readout trained with pixel-level supervision (Section 3.3). The second part asks whether localization ability is learned in the pretrained representation, in the adaptation, or in the prediction head, and provides two kinds of measurements to address this question: a criterion based on a shared linear readout for editing status across images (Section 3.4) and diagnostics that characterize how adaptation changes the representation (Section 3.5).

3.1 Dense foundation-model features

Localization requires spatially resolved features at every image position. We take them from DINOv3 (Siméoni et al., 2025), whose frozen patch features transfer well to dense prediction. Its vision Transformer processes a sequence of image-patch embeddings (Dosovitskiy et al., 2021). We evaluate ViT-S/16, ViT-B/16, and ViT-L/16. For an input of resolution $H \times W$, we remove the class and register tokens and take the final-layer, post-normalization patch tokens $\mathbf{F} = E_\theta(\mathbf{I}) \in \mathbb{R}^{N \times d}$, where $N = HW/16^2$ is the number of 16×16 patches. The same patch representation is used by all readouts and by the analyses in Sections 3.4 and 3.5.

3.2 Low-rank task adaptation

We adapt the encoder to make editing status more accessible to a simple shared readout while leaving the pretrained weights intact. In each attention block, LoRA (Hu et al., 2021) adds a trainable low-

rank update to the frozen fused query–key–value projection $\mathbf{W}$: $\mathbf{W}' = \mathbf{W} + \frac{\alpha}{r}\mathbf{AB}$, with rank $r = 32$ or 64 and $\alpha = 2r$. The LoRA matrices are trained jointly with the prediction head. The frozen-backbone control omits LoRA and trains only the readout; full fine-tuning updates the entire backbone and the readout.

Because the updates enter both the query–key comparison and the value projection, adaptation can change how patches attend to one another as well as the content they exchange. For each attention head, the change in output decomposes exactly into a term due to changed attention weights, a term due to changed values, and their interaction. In Section 5, we retain only Q/K or only V updates and compare linear-probe AP after refitting the probe on each representation.

3.3 Spatial readout and supervision

A shallow readout exposes the localization capability of the adapted patch features with little additional spatial processing. The default head, Conv3, is a three-layer convolutional network that maps each patch feature to one logit; the logits are bilinearly upsampled to the input resolution and passed through a sigmoid. The linear-head variant replaces Conv3 with a single 1×1 projection. It classifies each adapted patch independently, so that all learned interactions between patches occur in the encoder. To test whether a more capable decoder can compare regions from fixed features, we also use a Transformer head with global context that projects patch features to 512 dimensions and applies two Transformer blocks with eight attention heads before a linear mask predictor (Appendix C). The benchmark models use pixel and boundary-weighted BCE loss (Ma et al., 2024a), in which each pixel is weighted by $1 + \lambda e_i$ with a ground-truth boundary indicator e_i; we use $\lambda = 20$ by default and $\lambda = 0$ for the BCE-only control and matched comparison. Full architecture details and trainable parameter counts are given in Appendix A.4.

3.4 Cross-image linear readability

To locate where localization ability is learned, we need a representation measure independent of the prediction head. Because manipulation evidence is relational, a useful forensic representation must express it consistently across images: a rule learned on some images should transfer to others. We therefore assess editing status with one linear rule shared across images and fitted independently of the mask head. For an image $\mathbf{I}$, let $\mathbf{F_I} \in \mathbb{R}^{N \times d}$ denote its patch features and let $\mathbf{y_I}$ denote the corresponding edited/authentic labels. We define ideal cross-image linear readability as

$$\mathcal{R}(\mathbf{F}) = \max_{\mathbf{w},b} \mathbb{E}_{\mathbf{I}}[\mathrm{AP}(\mathbf{y_I}, \mathbf{F_I w} + b\mathbf{1})] \,, \tag{1}$$

where the expectation weights datasets equally and images equally within each dataset. The criterion describes the best mean per-image AP attainable by one shared linear rule. A common linear family allows comparisons across representations. We use AP because it depends only on the ranking of scores and is therefore insensitive to calibration. In experiments, empirical readability is the test-set AP of a ridge probe (Appendix E).

Remark (global-affine invariance). If $\mathbf{F}_{1,\mathbf{I}} = \mathbf{F}_{0,\mathbf{I}}\mathbf{A} + \mathbf{1c}^\top$ for every image $\mathbf{I}$, with the same invertible $\mathbf{A}$ and $\mathbf{c}$, then $\mathcal{R}(\mathbf{F}_1) = \mathcal{R}(\mathbf{F}_0)$, since every linear score $\mathbf{F}_{1,\mathbf{I}}\mathbf{w} + b\mathbf{1}$ equals the linear score $\mathbf{F}_{0,\mathbf{I}}(\mathbf{Aw}) + (\mathbf{c}^\top\mathbf{w} + b)\mathbf{1}$ on $\mathbf{F}_0$, and conversely.

This invariance holds for the optimal linear rule; the fitted probe is its empirical counterpart.

3.5 Diagnosing representation change

Readability tells us whether editing status can be read from a representation, but not how adaptation makes it readable. We therefore use four diagnostics, each tied to a specific question. Let $\mathbf{F}_0$ and $\mathbf{F}_1$ denote the pretrained and adapted patch features, respectively.

Depth of readability. Where along the encoder does editing status become readable? Following the use of independent linear probes to inspect intermediate representations (Alain & Bengio, 2017),

we apply the readability measure of Section 3.4 to patch features after blocks 6, 12, 20, and 24, fitting a separate shared probe on the training features of each encoder at each selected depth.

Cross-image alignment. Is editing evidence expressed along a direction shared across images? For each encoder, let $\mathbf{c}_+$ and $\mathbf{c}_-$ be the mean edited and authentic token features over training images, and let $\mathbf{d} = \mathbf{c}_+ - \mathbf{c}_-$ be the source editing direction. For a test image with edited and authentic token means $\boldsymbol{\mu}_+$ and $\boldsymbol{\mu}_-$, we quantify alignment with the source direction as

$$a = \cos\big(\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-,\, \mathbf{d}\big)\,. \tag{2}$$

In each encoder's native feature coordinates, a high value indicates that edit–background directions are more closely aligned across images. The direction also defines a readout without a fitted classifier, which scores each token $\mathbf{f}$ by $\cos(\mathbf{f} - \mathbf{m}, \mathbf{d})$ with $\mathbf{m} = (\mathbf{c}_+ + \mathbf{c}_-)/2$. Its AP measures how effectively one class-mean direction ranks edited and authentic tokens.

Readout from residual directions. We estimate the principal directions of the adaptation residual $\Delta\mathbf{F} = \mathbf{F}_1 - \mathbf{F}_0$ on training images. We project the residual onto the leading k directions, fit a shared linear probe on the projected training features, and evaluate its AP on test images.

Dependence on the learned updates. Is the readability gain produced by the LoRA updates themselves? We scale all updates by $\beta \in [0, 1]$, or retain only the Q/K or only the V updates, and refit the probe on the resulting features. Refitting ensures that changes in readability reflect the representation rather than compatibility with a fixed head. All diagnostic details appear in Appendix E.

4 Benchmark Evaluation

4.1 Experimental Setup

Datasets. We evaluate on the tampered-image subsets of five benchmarks used in IMDL-BenCo (Ma et al., 2024b): **CASIAv1** (Dong et al., 2013) (920 images, splicing and copy-move), **Columbia** (Hsu & Chang, 2006) (180 uncompressed spliced images), **NIST16** (Guan et al., 2019) (564 manipulated images), **Coverage** (Wen et al., 2016) (100 copy-move images containing similar-but-genuine objects), and **IMD2020** (Novozámský et al., 2020) (2,010 real-world manipulated images). The image counts and evaluation masks follow the benchmark release.

Training protocols. We use the implementations of the CAT-Net and MVSS protocols released in IMDL-BenCo (Ma et al., 2024b). For the **CAT-Net protocol**, we use a tampered-image subset of the released training sources, comprising CASIAv2 (Dong et al., 2013), FantasticReality (Kniaz et al., 2019) (19,423 tampered images), IMD2020 (Novozámský et al., 2020), and tampCOCO (Kwon et al., 2022); evaluation uses CASIAv1, Columbia, NIST16, and Coverage. For the **MVSS protocol**, we train on CASIAv2 (Dong et al., 2013) alone (5,123 images) and evaluate on all five benchmarks.

Training recipe. Inputs are bilinearly resized to 512×512, or 448×448 for the backbone control in Appendix A.1. The reference configuration uses AdamW (learning rate 3×10^{-4}, weight decay 0.05), a 100-epoch cosine schedule with five warmup epochs, and FP16 training on one GPU. Accumulating ten 24-image batches gives an effective batch size of 240; the rank-64 ViT-L model under the MVSS protocol uses ten 16-image batches, an effective batch size of 160. Under the CAT-Net protocol, we sample 12,880 training examples per epoch, totaling 1,288,000; Appendix A details the training inventory, balanced sampling, and augmentation settings.

Evaluation and checkpoint selection. We use the IMDL-BenCo pixel-F1 evaluator (Ma et al., 2024b) with fixed threshold $p > 0.5$, averaging per-image F1 within each dataset and then equally across datasets. Each row for our models uses one checkpoint maximizing mean F1 across its protocol's target benchmarks, following the official IMDL-BenCo training scripts, the benchmark from which the prior results are taken. Appendix D reports best-versus-final results; Appendix F defines the spatial diagnostics. Prior scores and input sizes follow the sources in Tables 1 and 2.

Matched comparison. To separate the contributions of the encoder and the head, we train DINOv3-L with a frozen or LoRA-adapted encoder and with either Conv3 or the global Transformer head on CASIAv2, and add full fine-tuning with a backbone learning rate of 10^{-5} and layer-wise decay. All five models share the same data and 100-epoch schedule, are evaluated on the four CAT-Net test sets every five epochs, and are selected on 1,025 held-out CASIAv2 validation images (Appendix C).

4.2 Results under the IMDL-BenCo CAT-Net Protocol

Table 1: Pixel-level F1 under the IMDL-BenCo CAT-Net protocol. Prior results for MVSS-Net, PSCC-Net, CAT-Net, TruFor, and Mesorch are from Zhu et al. (2024); IML-ViT results are from Ma et al. (2024b). Our DINOv3 rows use ViT-L; ViT-S/B results are in Appendix B. Dark, medium, and light shades mark the top three per column: blue for datasets, amber for Avg. F1.

	Method	Size	CASIAv1	Columbia	NIST16	Coverage	**Avg. F1**
Prior work	MVSS-Net (Dong et al., 2023)	512×512	0.583	0.740	0.336	0.486	0.536
	PSCC-Net (Liu et al., 2022)	256×256	0.630	0.884	0.346	0.448	0.577
	CAT-Net (Kwon et al., 2022)	512×512	0.808	0.915	0.252	0.427	0.601
	TruFor (Guillaro et al., 2023)	512×512	0.818	0.885	0.348	0.457	0.627
	Mesorch (Zhu et al., 2024)	512×512	0.840	0.890	0.392	0.586	0.677
	IML-ViT (Ma et al., 2024a)	1024×1024	0.795	**0.948**	0.501	0.654	0.725
ViT-L	DINOv3 + LoRA r=32	512×512	0.907	0.941	**0.636**	**0.905**	**0.847**
	DINOv3 + LoRA r=64	512×512	**0.908**	0.927	0.633	0.882	0.837
	DINOv3 + Full Fine-tuning	512×512	0.882	0.938	0.616	0.866	0.826

Under the CAT-Net protocol, the ViT-L rank-32 model exceeds the reported macro-average F1 of IML-ViT by 12.2 percentage points, with scores of 0.847 and 0.725, respectively (Table 1). On NIST16 and Coverage, its scores are 0.636 and 0.905, versus 0.501 and 0.654 for IML-ViT. On Columbia, IML-ViT scores 0.948 versus 0.941 for our model while using twice the input resolution (1024×1024 for IML-ViT versus 512×512 for our model; Table 1).

4.3 Results under the IMDL-BenCo MVSS Protocol

Table 2: Pixel-level F1 under the IMDL-BenCo MVSS protocol. Prior results are from Ma et al. (2024b). Highlighting follows Table 1; ViT-S and ViT-B results are in Appendix B.

	Method	Size	Coverage	Columbia	NIST16	CASIAv1	IMD2020	**Avg. F1**
Prior work	Mantra-Net (Wu et al., 2019)	256×256	0.090	0.243	0.104	0.125	0.055	0.123
	MVSS-Net (Dong et al., 2023)	512×512	0.259	0.386	0.246	0.534	0.279	0.341
	CAT-Net (Kwon et al., 2022)	512×512	0.296	0.584	0.269	0.581	0.273	0.401
	ObjectFormer (Wang et al., 2022)	224×224	0.294	0.336	0.173	0.429	0.173	0.281
	PSCC-Net (Liu et al., 2022)	256×256	0.231	0.604	0.214	0.378	0.235	0.333
	NCL-IML (Zhou et al., 2023)	512×512	0.225	0.446	0.260	0.502	0.237	0.334
	TruFor (Guillaro et al., 2023)	512×512	0.419	0.865	0.324	0.721	0.322	0.530
	IML-ViT (Ma et al., 2024a)	1024×1024	0.435	0.780	0.331	0.721	0.327	0.519
ViT-L	DINOv3 + LoRA r=32	512×512	**0.822**	**0.943**	0.589	0.867	0.628	0.770
	DINOv3 + LoRA r=64	512×512	0.820	0.915	**0.621**	**0.873**	**0.641**	**0.774**
	DINOv3 + Full Fine-tuning	512×512	0.679	0.842	0.532	0.852	0.499	0.681

Under the MVSS protocol with CASIAv2-only training, ViT-L with rank-64 LoRA reaches 0.774 mean F1, compared with 0.530 for TruFor. Both ViT-L LoRA variants exceed the listed specialized methods on all five benchmarks evaluated under the MVSS protocol (Table 2).

Backbone scale and adaptation strategy. Under CAT-Net, performance increases from ViT-S to ViT-L for each adaptation setting, and LoRA exceeds full fine-tuning at every evaluated scale under both protocols with the shared optimization recipe (Appendix B). Lowering the backbone learning rate to 10^{-5} with layer-wise decay does not let full fine-tuning surpass LoRA either (Table 3). Ranks, layer coverage, DoRA, and AdaptFormer are compared in Appendices A.2 and A.5.

Table 3: Matched DINOv3-L models trained on CASIAv2. Mean F1 on the four CAT-Net test sets with source-validation (Src.) or target-benchmark (Bench.; Tables 1 and 2) checkpoint selection. LR: encoder learning rate; Params: trainable count.

Encoder	LR	Head	Params (M)	Src.	Bench.
Frozen	—	Conv3	5.90	0.3200	0.3321
Frozen	—	Transformer	8.93	0.4667	0.4676
LoRA r=32	10^{-4}	Conv3	9.05	**0.7853**	**0.7900**
LoRA r=32	10^{-4}	Transformer	12.08	0.7557	0.7714
Full FT	10^{-5}	Conv3	309.05	0.6676	0.6932

Table 4: Linear-probe AP at four encoder depths, with one shared probe fitted per encoder and depth.

Block	6	12	20	24
Frozen	0.304	0.430	0.521	0.501
Adapted	0.299	0.533	0.910	0.919

Robustness to post-processing. On CASIAv1, the ViT-L LoRA model has higher F1 than the compared methods at all tested Gaussian noise, Gaussian blur, and JPEG settings; at JPEG quality 50, it reaches 0.8060 versus 0.6552 for Mesorch (Table 20 and Appendix G).

5 Understanding the Forensic Readability Gap

The benchmark results show that a shallow head on an adapted DINOv3 encoder outperforms specialized IML methods. We test whether a stronger decoder closes the gap (Section 5.1), what adaptation changes (Sections 5.2 and 5.3), and why a shallow head suffices (Section 5.4). We call the encoders adapted jointly with the Conv3 and linear heads *Conv3-adapted* and *linear-adapted*.

5.1 Decoder Capacity versus Representation

Is the frozen-feature deficit a matter of decoder capacity? Table 3 tests this directly. A stronger decoder does help frozen features: replacing Conv3 with the global Transformer head raises mean F1 from 0.3200 to 0.4667. Adapting the encoder, however, yields a much larger gain with the same head, raising the Transformer configuration by 28.90 points to 0.7557, and the adapted encoder with Conv3 reaches 0.7853. Once the encoder is adapted, the global head brings no further gain. Full fine-tuning uses 34 times as many trainable parameters but reaches only 0.6676 with a backbone learning rate of 10^{-5}. The ordering is unchanged at the final epoch (Appendix C). Under the main CAT-Net recipe, the frozen encoder with Conv3 likewise reaches only 0.4410, versus 0.847 with LoRA (Table 6).

A matched readout study shows that the adaptation gain persists across readout families and training budgets. We train linear, token-wise MLP, and three-layer convolutional readouts, together with a convolutional control on spatially shuffled tokens, on frozen and on Conv3-adapted features, selecting learning rates and epochs on source validation only (Appendix E). Across all selected readouts, frozen features reach 0.47–0.53 token AP on the four CAT-Net test sets, whereas adapted features reach 0.91–0.92, and longer training does not meaningfully narrow the gap. Frozen features are still informative, since a constant score gives 0.14 AP: editing information is present but difficult to read consistently across images.

5.2 Cross-Image Alignment of Editing Evidence

We next ask what adaptation changes so that this information becomes readable across images. We measure empirical readability (Section 3.4) using a shared ridge probe across the four CAT-Net test sets. Readability rises from 0.5014 AP for frozen features to 0.9187 for the Conv3-adapted encoder and to 0.8953 for the linear-adapted encoder. The gain persists after standardizing or whitening features before fitting the probe (Table 16). It also extends to interior tokens more than 32 pixels from any edit boundary, where AP rises from 0.42 to 0.94 (Appendix E).

In the adapted encoder, this readability forms in middle-to-late blocks (Table 4). At block 6, frozen and adapted features are equally hard to read (0.30 AP). The gap opens between blocks 12 and 20, growing from 0.10 to 0.39, and the adapted encoder reaches 0.91 AP at block 20. No tested frozen depth exceeds 0.52 AP, so reading an intermediate frozen layer does not close the gap.

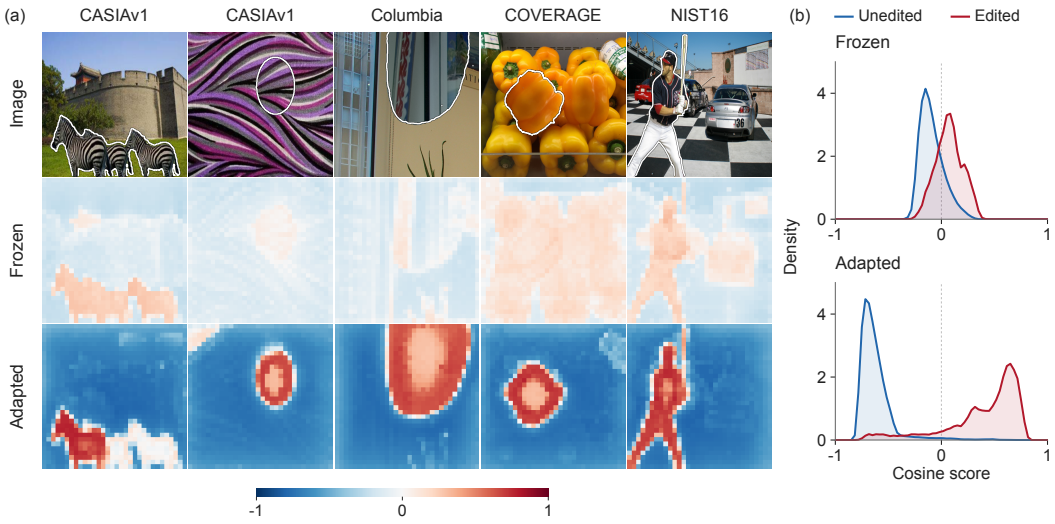

Figure 2: **Adaptation aligns edit–background separation across images.** (a) Cosine-score maps for five pre-selected examples from four datasets, including splicing and copy-move. Each encoder uses one editing direction estimated from source class means and fixed across all test images; token features are centered at the midpoint of these means. White contours on input images mark ground-truth edits, and all heatmaps share the $[-1, 1]$ scale. (b) Distributions of the same score over all 1,751 test images with both token classes. Each class is normalized within each image, images are averaged within datasets, and datasets are weighted equally. Adapted features are from the encoder trained jointly with the three-layer convolutional head.

Table 5: Cross-image readability and direction alignment by test set and manipulation type, for frozen and Conv3-adapted features. AP is measured with a linear probe. Gap closed is $(\mathrm{AP}_{\mathrm{adapted}} - \mathrm{AP}_{\mathrm{frozen}})/(1 - \mathrm{AP}_{\mathrm{frozen}})$. Alignment is the mean cosine between each image's edited-minus-authentic feature difference and the source editing direction. CASIAv1 subsets use the splicing and copy-move directories; NIST16 contains mixed manipulations. The Images column counts CAT-Net test images with both token classes, as required for AP and alignment.

		Token AP			Alignment	
Test set	Images	Frozen	Adapted	Gap closed	Frozen	Adapted
CASIAv1	919	0.595	0.961	90%	0.231	0.867
splicing	460	0.796	0.984	92%	0.333	0.897
copy-move	459	0.393	0.939	90%	0.129	0.838
Columbia (splicing)	180	0.554	0.958	91%	0.168	0.710
COVERAGE (copy-move)	100	0.332	0.949	92%	0.218	0.861
NIST16 (mixed)	552	0.525	0.807	59%	0.229	0.696
Macro average	1,751	0.501	0.919	84%	0.211	0.784

The readout gain is accompanied by stronger cross-image direction alignment. Using the source editing direction $\mathbf{d}$ of Section 3.5, we measure the alignment a of each test image (Equation (2)). In frozen features this cosine averages only 0.21: the direction that separates edited regions from their background changes from image to image. After adaptation, it averages 0.78 for both adapted encoders, an increase of 0.57 [0.56, 0.58] for the Conv3-adapted encoder (paired image bootstrap).

A readout built from this direction alone performs almost as well as the fitted probe. Scoring each token by its cosine to $\mathbf{d}$, measured from the midpoint of the two class means and without any fitted classifier, gives 0.4929 AP on frozen features and 0.9155 on Conv3-adapted features, almost identical to the fitted probe. Thresholding this score at zero yields a token-level F1 of 0.8474 after adaptation, compared with 0.4206 before. Figure 2 visualizes the score: before adaptation, edited

and authentic tokens overlap around zero; afterwards, they fall on opposite sides of the same fixed boundary for test images from all four CAT-Net datasets.

The effect holds across manipulation types (Table 5). Within CASIAv1, frozen features are far more readable on spliced images (0.80 AP) than on copy-move images (0.39), yet adaptation brings both above 0.93 and raises their alignment from 0.33 and 0.13 to 0.90 and 0.84. A controlled copy-move experiment with pixel-identical source and target interiors shows much stronger separation of editing roles after adaptation (role AUC 0.99 versus 0.70; Appendix H).

Affine controls also show a large empirical gap. A global affine map fitted from frozen to adapted features reaches 0.5217 AP, and a linear map fitted to reproduce the adapted scores reaches 0.5128. Both remain close to the frozen-feature probe at 0.5014 AP.

5.3 Sources of the Change

Two interventions connect the readability gain to the LoRA updates. First, we scale all updates of the linear-adapted encoder by $\beta \in \{0, 0.25, 0.5, 0.75, 1\}$ and refit the probe at each strength. Readability increases at every step, from 0.5014 to 0.6088, 0.7632, 0.8787, and 0.8953. Second, we retain only the Q/K updates or only the V updates in all 24 blocks of the Conv3-adapted encoder and refit the probe. Readability drops from 0.9187 to 0.7072 and 0.7990, respectively. V-only exceeds Q/K-only in every manipulation group; retaining both pathways gives the highest readout AP.

The adaptation residual $\Delta\mathbf{F} = \mathbf{F}_1 - \mathbf{F}_0$ is broad: explaining 90% of its variance requires 472 directions for the Conv3-adapted encoder and 493 for the linear-adapted encoder. Yet a probe on its first 16 principal directions retains 0.9025 and 0.8488 AP (Table 17). A few shared directions therefore suffice for strong discrimination, but the evidence is not confined to them: combining the frozen features with the residual outside these directions still yields 0.9050 and 0.8952.

5.4 Role of the Mask Head

With an adapted encoder, the head needs little additional capacity: a linear head already reaches 0.8145, and the jointly trained Conv3 model adds 3.25 points. We compare the two models on the same 1,764 test images at $p > 0.5$ on a common 512×512 grid (Appendix F). The Conv3 model has 4.74 points higher boundary F1 at a two-pixel tolerance, 3.60 points higher F1 outside a two-pixel band around the contour, and a 3.04-point lower false-negative rate on edited pixels outside that band. Conv3 thus localizes both the interior and the boundary of edited regions better.

The tested decoder extensions perform similarly to Conv3. At 448×448 input, subpixel upsampling, a residual local-refinement block, and a progressive decoder that fuses block-20 features change mean F1 by -0.27 to $+0.04$ points relative to Conv3 under the same adaptation and training budget (Table 9 and Appendix A.3). In the matched comparison, the global Transformer head likewise does not improve on Conv3 once the encoder is adapted (0.7557 versus 0.7853; Table 3).

6 Conclusion

Frozen foundation features are a natural starting point for IML, yet their limitation lies less in the decoder than in the representation. A stronger decoder improves frozen DINOv3 features, but low-rank encoder adaptation with a comparable total number of trainable parameters yields a far larger gain, and full fine-tuning with 34 times as many trainable parameters does not surpass it. The adapted representation expresses editing evidence consistently across images: shared linear-probe AP rises from 0.50 to 0.92, editing directions become aligned, and a few shared directions suffice for strong discrimination. This structure forms in middle-to-late blocks after adaptation, whereas no tested frozen layer exceeds 0.52 AP. On this representation, a shallow head is enough: without a dedicated forensic front end, DINOv3-IML outperforms specialized methods by 12.2 and 24.4 points under the CAT-Net and MVSS protocols and sets a simple yet strong baseline. These results suggest that for forensic localization, where trainable capacity is placed matters more than how much is added.

Reproducibility statement

The architecture and training recipe are described in Sections 3 and 4.1. Full per-dataset benchmark results are in Tables 1 and 2. The appendices provide adaptation settings, parameter counts, checkpoint selection and final-epoch comparisons, and the definitions and uncertainty estimates for the paired diagnostics. Code for the core implementation is available at:

https://anonymous.4open.science/r/forensic-readability-C062/.

AI use statement

Generative AI tools assisted with language editing, drafting and revising portions of the manuscript, literature and evaluation-protocol cross-checking, research planning, feedback on methodology and experimental analysis, reviewer-style critique, and LaTeX and code formatting. The authors take responsibility for the final content and verification of citations, implementation, results, and claims.

Ethics statement

Image manipulation localization can support media-integrity investigations, but model predictions may be unreliable under distribution shift and should not be treated as stand-alone evidence about provenance or intent. We report aggregate benchmark results and limitations to discourage such use. This work uses only existing research datasets and does not introduce new personal data.

References

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *International Conference on Learning Representations, Workshop Track*, 2017. URL https://openreview.net/forum?id=HJ4-rAVtl.

Belhassen Bayar and Matthew C. Stamm. A deep learning approach to universal image manipulation detection using a new convolutional layer. In *Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security*, pp. 5–10, 2016. doi: 10.1145/2909827.2930786. URL https://doi.org/10.1145/2909827.2930786.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers, 2021. URL https://arxiv.org/abs/2104.14294v2.

Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. AdaptFormer: Adapting vision transformers for scalable visual recognition. In *Advances in Neural Information Processing Systems*, volume 35, pp. 16664–16678, 2022. URL https://papers.nips.cc/paper_files/paper/2022/hash/69e2f49ab0837b71b0e0cb7c555990f8-Abstract-Conference.html.

Davide Cozzolino and Luisa Verdoliva. Noiseprint: a CNN-based camera model fingerprint, 2018. URL https://arxiv.org/abs/1808.08396v1.

Minh Thong Doi, Vincent Itier, Jan Butora, Jérémie Boulanger, and Patrick Bas. DinoLizer: Separating VAE and diffusion artifacts in generative inpainting localization, 2026. URL https://arxiv.org/abs/2511.20722v2.

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, March 2023. ISSN 1939-3539. doi: 10.1109/tpami.2022.3180556. URL https://doi.org/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. IEEE, July 2013. doi: 10.1109/chinasip.2013.6625374. URL https://doi.org/10.1109/chinasip.2013.6625374.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Haiying Guan, Mark Kozak, Eric Robertson, Yooyoung Lee, Amy N. Yates, Andrew Delgado, Daniel Zhou, Timothee Kheyrkhah, Jeff Smith, and Jonathan Fiscus. MFC datasets: Large-scale benchmark datasets for media forensic challenge evaluation. In *2019 IEEE Winter Applications of Computer Vision Workshops (WACVW)*, pp. 63–72, 2019. doi: 10.1109/WACVW.2019.00018. URL https://doi.org/10.1109/WACVW.2019.00018.

Fabrizio Guillaro, Davide Cozzolino, Avneesh Sud, Nicholas Dufour, and Luisa Verdoliva. TruFor: Leveraging all-round clues for trustworthy image forgery detection and localization, 2023. URL https://arxiv.org/abs/2212.10957v3.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross B. Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15979–15988, 2022. doi: 10.1109/CVPR52688.2022.01553. URL https://doi.org/10.1109/CVPR52688.2022.01553.

Yu-feng Hsu and Shih-fu Chang. Detecting image splicing using geometry invariants and camera characteristics consistency. In *2006 IEEE International Conference on Multimedia and Expo*, pp. 549–552, 2006. doi: 10.1109/ICME.2006.262447. URL https://doi.org/10.1109/ICME.2006.262447.

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, 2021. URL https://arxiv.org/abs/2106.09685v2.

Xuefeng Hu, Zhihan Zhang, Zhenye Jiang, Syomantak Chaudhuri, Zhenheng Yang, and Ram Nevatia. SPAN: Spatial pyramid attention network for image manipulation localization. In *Computer Vision – ECCV 2020*, pp. 312–328, 2020. doi: 10.1007/978-3-030-58589-1_19. URL https://doi.org/10.1007/978-3-030-58589-1_19.

Minyoung Huh, Andrew Liu, Andrew Owens, and Alexei A. Efros. Fighting fake news: Image splice detection via learned self-consistency. In *Computer Vision – ECCV 2018*, pp. 106–124, 2018. doi: 10.1007/978-3-030-01252-6_7. URL https://doi.org/10.1007/978-3-030-01252-6_7.

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge J. Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *Computer Vision – ECCV 2022*, pp. 709–727, 2022. doi: 10.1007/978-3-031-19827-4_41. URL https://doi.org/10.1007/978-3-031-19827-4_41.

Vladimir V. Kniaz, Vladimir Knyaz, and Fabio Remondino. The point where reality meets fantasy: Mixed adversarial generators for image splice detection. In *NeurIPS*, pp. 215–226, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/98dce83da57b0395e163467c9dae521b-Abstract.html.

Myung-Joon Kwon, Seung-Hun Nam, In-Jae Yu, Heung-Kyu Lee, and Changick Kim. Learning JPEG compression artifacts for image manipulation detection and localization. *International Journal of Computer Vision*, 130(8):1875–1895, May 2022. ISSN 1573-1405. doi: 10.1007/s11263-022-01617-5. URL https://doi.org/10.1007/s11263-022-01617-5.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. DoRA: Weight-decomposed low-rank adaptation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32100–32121. PMLR, 2024. URL https://proceedings.mlr.press/v235/liu24bn.html.

Xiaohong Liu, Yaojie Liu, Jun Chen, and Xiaoming Liu. PSCC-Net: Progressive spatio-channel correlation network for image manipulation detection and localization, 2022. URL https://arxiv.org/abs/2103.10596v2.

Xiaochen Ma, Bo Du, Zhuohang Jiang, Xia Du, Ahmed Y. Al Hammadi, and Jizhe Zhou. IML-ViT: Benchmarking image manipulation localization by vision transformer, 2024a. URL https://arxiv.org/abs/2307.14863v4.

Xiaochen Ma, Xuekang Zhu, Lei Su, Bo Du, Zhuohang Jiang, Bingkui Tong, Zeyu Lei, Xinyu Yang, Chi-Man Pun, Jiancheng Lv, and Jizhe Zhou. IMDL-BenCo: A comprehensive benchmark and codebase for image manipulation detection & localization, 2024b. URL https://arxiv.org/abs/2406.10580v2.

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, 2025. URL https://arxiv.org/abs/2506.20922v1.

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. URL https://doi.org/10.1109/WACVW50321.2020.9096940.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision, 2024. URL https://arxiv.org/abs/2304.07193v2.

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/2508.10104v1.

Luisa Verdoliva. Media forensics and deepfakes: An overview. *IEEE Journal of Selected Topics in Signal Processing*, 14(5):910–932, August 2020. ISSN 1941-0484. doi: 10.1109/jstsp.2020.3002101. URL https://doi.org/10.1109/JSTSP.2020.3002101.

Junke Wang, Zuxuan Wu, Jingjing Chen, Xintong Han, Abhinav Shrivastava, Ser-Nam Lim, and Yu-Gang Jiang. ObjectFormer for image manipulation detection and localization, 2022. URL https://arxiv.org/abs/2203.14681v2.

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. URL https://doi.org/10.1109/ICIP.2016.7532339.

Yue Wu, Wael Abd-Almageed, and Premkumar Natarajan. ManTra-Net: Manipulation tracing network for detection and localization of image forgeries with anomalous features. In *CVPR*, pp. 9543–9552, 2019. URL https://openaccess.thecvf.com/content_CVPR_2019/html/Wu_ManTra-Net_Manipulation_Tracing_Network_for_Detection_and_Localization_of_Image_CVPR_2019_paper.html.

Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan L. Yuille, and Tao Kong. Image BERT pre-training with online tokenizer. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=ydopy-e6Dg.

Jizhe Zhou, Xiaochen Ma, Xia Du, Ahmed Y. Alhammadi, and Wentao Feng. Pre-training-free image manipulation localization through non-mutually exclusive contrastive learning, 2023. URL https://arxiv.org/abs/2309.14900v2.

Xuekang Zhu, Xiaochen Ma, Lei Su, Zhuohang Jiang, Bo Du, Xiwen Wang, Zeyu Lei, Wentao Feng, Chi-Man Pun, and Jizhe Zhou. Mesoscopic insights: Orchestrating multi-scale & hybrid architecture for image manipulation localization, 2024. URL https://arxiv.org/abs/2412.13753v1.

A Component and Backbone Studies: Detailed Results

Table 6: Component configurations under the CAT-Net protocol. Δ is the mean-F1 difference from the full model in percentage points. All use DINOv3-L except the DINOv2 row. The linear head retains edge supervision; backbones are compared at 448 input resolution.

Configuration	Adaptation	Head	λ	Input	Mean F1	Δ
Full model	LoRA $r = 32$	Conv3	20	512	0.847	—
Frozen encoder	Frozen	Conv3	20	512	0.4410	-40.60
Linear head	LoRA $r = 32$	Linear	20	512	0.8145	-3.25
BCE only	LoRA $r = 32$	Conv3	0	512	0.8328	-1.42
DINOv3 (448)	LoRA $r = 32$	Conv3	20	448	0.8333	—
DINOv2 (448)	LoRA $r = 32$	Conv3	20	448	0.8072	—

Table 7: Per-dataset results for the component and backbone variants under the IMDL-BenCo CAT-Net protocol. Mean F1 is computed from unrounded dataset scores.

Configuration	CASIAv1	Columbia	NIST16	Coverage	Mean F1
Frozen encoder	0.6152	0.6289	0.3283	0.1914	0.4410
Linear head	0.8849	0.9419	0.5955	0.8356	0.8145
BCE only	0.8992	0.9312	0.6237	0.8773	0.8328
DINOv3 (448)	0.8941	0.9436	0.6165	0.8789	0.8333
DINOv2 (448)	0.9044	0.9022	0.5781	0.8440	0.8072

Data and sampling. The CAT-Net training inventory contains 825,997 entries: 5,123 CASIAv2, 19,423 FantasticReality, 2,010 IMD2020, and four tampCOCO subsets with 199,999, 199,999, 200,000, and 199,443 entries. The balanced sampler draws 1,840 samples per source per epoch, or 12,880 samples in total. Optimization follows the reference configuration in Section 4.1. Training augmentations include random scaling, synthetic copy-move and inpainting, horizontal and vertical flips, brightness and contrast changes, JPEG compression, rotations by multiples of $90°$, and Gaussian blur, followed by resizing and normalization in the IMDL-BenCo pipeline.

Variant definitions. The full DINOv3-L model uses rank-32 LoRA, Conv3, and edge supervision at 512×512, reusing the full-model checkpoint reported in Table 1. Other variants are trained separately. The frozen-encoder variant trains only Conv3; the linear-head variant retains edge supervision, and the BCE-only variant omits it. The backbone comparison uses DINOv3-L/16 and DINOv2-L/14 at 448×448; their 2.61-point mean-F1 difference is computed from unrounded scores.

A.1 Supporting Backbone Control at Matched Input Resolution

At 448×448 with the same recipe, DINOv3-L/16 reaches 0.8333 mean F1 and DINOv2-L/14 reaches 0.8072, a difference of 2.61 percentage points. The gains are 4.14 points on Columbia, 3.84 on NIST16, and 3.48 on Coverage; CASIAv1 decreases from 0.9044 to 0.8941 (Table 7). DINOv3 uses a 28×28 patch grid at this resolution and DINOv2 a 32×32 grid. The comparison supports a mean-F1 advantage for the released DINOv3 backbone under this recipe, despite its coarser grid.

A.2 Adaptation Rank and Layer Coverage

The low-rank variants use rank 8 and rank 16 in the fused QKV projections of all 24 DINOv3-L blocks. The late-only variant uses rank 32 in blocks 17–24. All three read final-layer features with Conv3 at 512×512, use LoRA scale $\alpha = 2r$, and retain edge weight 20. They follow the data inventory, balanced sampling, optimizer, and 100-epoch schedule described above, each with seed 42. The original pretrained weights remain frozen in all three configurations.

Performance is stable across ranks: rank 8 across all 24 blocks reaches 0.8360 mean F1 with 0.786M adaptation parameters, one quarter of the adaptation parameters used at rank 32. Adapting only the

Table 8: Adaptation rank and layer coverage with Conv3 at 512 input resolution. LoRA and total trainable parameter counts are in millions. The all-block rank-32 and rank-64 entries reproduce the ViT-L results in Table 1; the lower-rank and late-only configurations are additional runs.

Adapted blocks	Rank	LoRA (M)	Total (M)	Mean F1
All 24	8	0.7864	6.6872	0.8360
All 24	16	1.5729	7.4737	0.8295
All 24	32	3.1457	9.0465	0.847
All 24	64	6.2915	12.1923	0.837
Last 8	32	1.0486	6.9494	0.7998

last eight blocks at rank 32 reaches 0.7998 with 1.049M adaptation parameters. The all-block rank-8 configuration exceeds late-only adaptation by 3.62 percentage points with fewer LoRA parameters.

A.3 Decoder Extensions at Matched Resolution

The subpixel, residual-refinement, and intermediate-feature decoders start from the same pretrained DINOv3-L weights as the 448-input Conv3 baseline and are trained independently for 100 epochs with seed 42. Their shared settings include 448×448 inputs, rank-32 QKV LoRA in all 24 blocks, $\alpha = 64$, the same training inventory and balanced sampler, batch size and accumulation, optimizer, augmentation, and edge-supervision settings. Only the decoder architecture changes. At this input resolution, the encoder produces a final patch grid of 28×28.

Table 9: Decoder extensions with the same rank-32 DINOv3-L adaptation at 448 input resolution. Δ is the mean-F1 difference from Conv3 in percentage points. Parameter counts cover the decoder only; each encoder has 3.146M trainable LoRA parameters.

Readout	Parameters (M)	Mean F1	Δ
Conv3	5.9008	0.8333	—
Subpixel upsampling	5.9047	0.8337	$+0.04$
Residual local refinement	6.4918	0.8305	-0.27
Intermediate-feature fusion	6.0754	0.8312	-0.20

Subpixel upsampling. The two spatial convolutions of Conv3 are retained. A final projection predicts 16 channels per token, which are rearranged into a 112×112 single-channel logit map and then bilinearly upsampled to the input size. This adds 3,855 parameters relative to Conv3.

Residual local refinement. A residual branch applies two 3×3 convolutions with channel dimensions $256 \to 128 \to 256$ before the final output projection. The last normalization scale is initialized to zero, giving an initial identity refinement. The branch adds 590,976 parameters.

Intermediate-feature fusion. Features from block 20 and the 256-channel hidden map of Conv3 are each projected to 64 channels and fused at 56×56. Two stages of upsampling and spatial convolution produce a 112×112 correction, which is upsampled and added to the baseline logits. The output correction is initialized to zero. The block-20 features have the same token grid as the final features; the added branch learns the finer readout and adds 174,561 parameters. Gradients from both branches update the shared LoRA parameters.

A.4 Readout, Supervision, and Parameter Counts

Feature extraction and readout. The decoder reads the final normalized patch features; class and register tokens are excluded. Conv3 uses two padded 3×3 convolutions and one 1×1 convolution, with channel widths $d \to d/2 \to d/4 \to 1$, biases in all three layers, and affine batch normalization after each spatial convolution. ReLU follows batch normalization. The logits are bilinearly upsampled to the input resolution before applying the sigmoid. The linear variant uses a biased

1×1 convolution and the same upsampling. The frozen backbone uses the same feature-extraction settings during training and evaluation.

Boundary weighting. The benchmark first applies its common image/mask transforms, converts the GT to a binary mask, and constructs a boundary-weight map using a cross-shaped kernel of size 7. Let D_7 denote binary dilation by this kernel with zero padding. For a binary mask M, the map is $e = D_7(M)D_7(1-M)$, the locations whose cross-shaped neighborhoods intersect both foreground and background. The map and GT are then resized jointly by the dataset's post-transform. Thus the kernel parameter is not a seven-pixel Euclidean contour tolerance on the final input grid. The loss is the all-pixel mean of BCE plus 20 times the all-pixel mean of edge-weighted BCE; the edge contribution has no separate normalization by edge area. This training map is distinct from the two-pixel boundary band used only for the diagnostic metrics in Appendix F.

Table 10: Analytical trainable parameter counts for the fused QKV LoRA and Conv3 layers, in millions. Counts include convolution biases and affine BN parameters, and exclude frozen weights and BN running statistics. Rank-64 totals use the same Conv3 head.

Backbone	Blocks	LoRA $r = 32$	Conv3	Total $r = 32$	Total $r = 64$
ViT-S	12	0.589824	0.830401	1.420225	2.010049
ViT-B	12	1.179648	3.319681	4.499329	5.678977
ViT-L	24	3.145728	5.900801	9.046529	12.192257

For embedding dimension d, L blocks, and rank r, the LoRA parameter count is $4Ldr$ for the fused $d \to 3d$ QKV projection. The ViT-L linear head has 1,025 parameters, giving 3,146,753 trainable parameters together with rank-32 LoRA. These totals include the trainable adaptation and decoder parameters; frozen backbone weights are excluded.

A.5 Additional Adaptation Configurations

We additionally evaluate DoRA (Liu et al., 2024) and an AdaptFormer-style configuration (Chen et al., 2022) on DINOv3-L at 512×512. Both use Conv3, BCE with edge weight 20, and the same 100-epoch training recipe under the CAT-Net protocol as the component runs.

Table 11: Additional adaptation configurations under the CAT-Net protocol, with the full LoRA model as reference. Scores are pixel F1, with the macro-average computed before rounding.

Configuration	CASIAv1	Columbia	NIST16	Coverage	Mean F1
LoRA $r = 32$	0.907	0.941	0.636	0.905	0.847
DoRA $r = 32$	0.9071	0.9504	0.6277	0.8992	0.8461
AdaptFormer-style	0.8998	0.9317	0.6128	0.8349	0.8198

Configuration details. DoRA updates the fused QKV projection in all 24 blocks, using rank 32 and $\alpha = 64$ with a learned magnitude per output channel. Its QKV computations use activation recomputation to fit the same physical batch size. The AdaptFormer-style model uses one parallel $1024 \to 64 \to 1024$ FFN adapter per block, with ReLU, dropout 0.1, fixed scale 0.1, no added normalization, and zero-initialized up projections. The frozen original FFN and LayerScale remain in place, and LoRA is not combined with these adapters. The adapter branch has 3,171,840 trainable parameters, in addition to the 5,900,801 parameters in Conv3.

B Results Across Backbone Scales

Tables 12 and 13 report ViT-S, ViT-B, and ViT-L results with LoRA ranks 32 and 64 and with full fine-tuning under the CAT-Net and MVSS protocols, respectively.

Table 12: Pixel-level F1 under the IMDL-BenCo CAT-Net protocol. Prior results for MVSS-Net, PSCC-Net, CAT-Net, TruFor, and Mesorch are from Zhu et al. (2024); IML-ViT results are from Ma et al. (2024b). All DINOv3 rows are our models, for three backbone scales. Dark, medium, and light blue mark the top three results per column across all methods.

	Method	Size	CASIAv1	Columbia	NIST16	Coverage	Avg. F1
Prior work	MVSS-Net (Dong et al., 2023)	512×512	0.583	0.740	0.336	0.486	0.536
	PSCC-Net (Liu et al., 2022)	256×256	0.630	0.884	0.346	0.448	0.577
	CAT-Net (Kwon et al., 2022)	512×512	0.808	0.915	0.252	0.427	0.601
	TruFor (Guillaro et al., 2023)	512×512	0.818	0.885	0.348	0.457	0.627
	Mesorch (Zhu et al., 2024)	512×512	0.840	0.890	0.392	0.586	0.677
	IML-ViT (Ma et al., 2024a)	1024×1024	0.795	**0.948**	0.501	0.654	0.725
ViT-S	DINOv3 + LoRA r=32	512×512	0.787	0.923	0.462	0.646	0.704
	DINOv3 + LoRA r=64	512×512	0.803	0.918	0.457	0.671	0.712
	DINOv3 + Full Fine-tuning	512×512	0.704	0.887	0.400	0.531	0.630
ViT-B	DINOv3 + LoRA r=32	512×512	0.840	0.938	0.565	0.715	0.764
	DINOv3 + LoRA r=64	512×512	0.863	0.904	0.570	0.784	0.780
	DINOv3 + Full Fine-tuning	512×512	0.815	0.918	0.518	0.652	0.726
ViT-L	DINOv3 + LoRA r=32	512×512	0.907	0.941	**0.636**	**0.905**	**0.847**
	DINOv3 + LoRA r=64	512×512	**0.908**	0.927	0.633	0.882	0.837
	DINOv3 + Full Fine-tuning	512×512	0.882	0.938	0.616	0.866	0.826

Table 13: Pixel-level F1 under the IMDL-BenCo MVSS protocol. Prior results are from Ma et al. (2024b). Highlighting follows Table 12; all three backbone scales are shown.

	Method	Size	Coverage	Columbia	NIST16	CASIAv1	IMD2020	Avg. F1
Prior work	Mantra-Net (Wu et al., 2019)	256×256	0.090	0.243	0.104	0.125	0.055	0.123
	MVSS-Net (Dong et al., 2023)	512×512	0.259	0.386	0.246	0.534	0.279	0.341
	CAT-Net (Kwon et al., 2022)	512×512	0.296	0.584	0.269	0.581	0.273	0.401
	ObjectFormer (Wang et al., 2022)	224×224	0.294	0.336	0.173	0.429	0.173	0.281
	PSCC-Net (Liu et al., 2022)	256×256	0.231	0.604	0.214	0.378	0.235	0.333
	NCL-IML (Zhou et al., 2023)	512×512	0.225	0.446	0.260	0.502	0.237	0.334
	TruFor (Guillaro et al., 2023)	512×512	0.419	0.865	0.324	0.721	0.322	0.530
	IML-ViT (Ma et al., 2024a)	1024×1024	0.435	0.780	0.331	0.721	0.327	0.519
ViT-S	DINOv3 + LoRA r=32	512×512	0.363	0.646	0.324	0.622	0.346	0.460
	DINOv3 + LoRA r=64	512×512	0.403	0.721	0.343	0.672	0.373	0.502
	DINOv3 + Full Fine-tuning	512×512	0.154	0.364	0.189	0.209	0.186	0.221
ViT-B	DINOv3 + LoRA r=32	512×512	0.211	0.443	0.256	0.543	0.296	0.350
	DINOv3 + LoRA r=64	512×512	0.545	0.820	0.465	0.761	0.475	0.613
	DINOv3 + Full Fine-tuning	512×512	0.163	0.136	0.093	0.078	0.082	0.110
ViT-L	DINOv3 + LoRA r=32	512×512	**0.822**	**0.943**	0.589	0.867	0.628	0.770
	DINOv3 + LoRA r=64	512×512	0.820	0.915	**0.621**	**0.873**	**0.641**	**0.774**
	DINOv3 + Full Fine-tuning	512×512	0.679	0.842	0.532	0.852	0.499	0.681

C Matched Encoder and Head Comparison

All five matched models in Table 3 use DINOv3 ViT-L/16 with 512×512 inputs and are trained on 4,098 CASIAv2 images for 100 epochs with pixel BCE, no augmentation, batch size 16, and AdamW (weight decay 0.01), using a short warm-up followed by a two-stage cosine schedule. Checkpoints are saved every five epochs and selected by mean F1 on 1,025 held-out CASIAv2 validation images. LoRA uses rank 32 and $\alpha = 64$ with a backbone learning rate of 10^{-4}. The head learning rate is 10^{-3} for Conv3 and 3×10^{-4} for the Transformer head, the latter chosen on source validation with the frozen encoder and used for both encoders. Full fine-tuning uses a backbone learning rate of 10^{-5} with a layer-wise learning-rate decay factor of 0.85.

Transformer head. A linear layer projects the 1024-dimensional patch features to 512 dimensions and adds fixed two-dimensional sinusoidal position encodings. Two pre-norm Transformer blocks with eight attention heads, a 3072-dimensional feed-forward layer, and GELU activations follow, and a LayerNorm with a linear layer predicts one logit per patch, which is bilinearly upsampled. The Transformer head has 8.93M trainable parameters.

Per-dataset and final-epoch results. Table 14 reports per-dataset F1 and the mean F1 at the final (100th) epoch. The final epoch gives the same ordering of the five models.

Table 14: Per-dataset pixel F1 of the matched encoder and head comparison with source-validation checkpoint selection, and mean F1 at the final epoch.

Encoder	Head	CASIAv1	COVERAGE	Columbia	NIST16	Mean	Final
Frozen	Conv3	0.5507	0.2142	0.2560	0.2593	0.3200	0.3195
Frozen	Transformer	0.6329	0.2784	0.5319	0.4237	0.4667	0.4570
LoRA	Conv3	0.8630	0.7854	0.8760	0.6167	0.7853	0.7794
LoRA	Transformer	0.8554	0.7178	0.8337	0.6160	0.7557	0.7526
FT, 10^{-5}	Conv3	0.8340	0.6193	0.6774	0.5397	0.6676	0.6642

D Checkpoint Selection and Final-Epoch Sensitivity

Selection rule. For each configuration, we select the checkpoint with the highest equal-dataset mean pixel-level F1 over the evaluation datasets of its protocol: four datasets for the CAT-Net protocol and five for the MVSS protocol. All dataset scores in a row come from this single checkpoint. Selection and reporting use the same fixed threshold $p > 0.5$. This follows the official IMDL-BenCo training scripts. The matched comparison in Table 3 instead selects checkpoints on source validation (Appendix C). Epochs denote completed training epochs, numbered from 1 to 100.

Final-epoch comparison. For the component, backbone, rank, decoder, and additional adaptation configurations in Table 15, evaluation takes place after epochs $1, 5, \ldots, 97$, and 100, giving 26 evaluations per run. We compare the checkpoint selected by mean F1 under the CAT-Net protocol with the fixed epoch-100 endpoint to assess sensitivity to checkpoint selection.

Table 15: Best-selected versus final-epoch mean F1 under the CAT-Net protocol for the component, backbone, rank, decoder, and additional adaptation studies. All final scores are from epoch 100; Δ is best minus final in percentage points, computed before rounding.

Configuration	Input	Best F1	Final F1	Δ
Frozen encoder	512	0.4410	0.4248	1.62
Linear head	512	0.8145	0.8130	0.15
BCE only	512	0.8328	0.8255	0.74
DINOv3 (448)	448	0.8333	0.8261	0.72
DINOv2 (448)	448	0.8072	0.8053	0.19
All 24 blocks, rank 8	512	0.8360	0.8335	0.26
All 24 blocks, rank 16	512	0.8295	0.8248	0.47
Last 8 blocks, rank 32	512	0.7998	0.7979	0.19
Subpixel upsampling	448	0.8337	0.8224	1.13
Residual local refinement	448	0.8305	0.8239	0.67
Intermediate-feature fusion	448	0.8312	0.8254	0.58
DoRA rank 32	512	0.8461	0.8365	0.96
AdaptFormer-style	512	0.8198	0.8175	0.23

Sensitivity of the findings. At epoch 100, the linear head retains 0.8130 mean F1 and all-block rank-8 adaptation retains 0.8335, exceeding late-only adaptation by 3.55 percentage points. The three decoder extensions remain below Conv3, by 0.07–0.37 points. Thus the linear-readout and low-rank results, and the limited benefit of these decoder extensions, persist at the final epoch. The supporting DINOv3–DINOv2 gap is 2.08 points at this endpoint.

E Representation Diagnostics: Implementation Details

Features and checkpoints. Except for the depth analysis in Table 4, diagnostics use final-layer, post-normalization patch tokens on the 32×32 grid of a 512×512 input. Pretrained features come from the original DINOv3 ViT-L/16 encoder; adapted features come from the Conv3-adapted and linear-adapted encoders of Section 5. No encoder is trained during these diagnostics; only the specified update interventions modify its weights.

Source data and token sampling. We split the 5,123 CASIAv2 images into 4,098 training and 1,025 source-validation images. Readouts, class means, and principal components are fitted on the 3,932 training images with eligible tokens. From each eligible training image we sample 64 edited tokens, covering at least 75% edited pixels, and 64 authentic tokens, covering no edited pixels. The same token indices are reused for all encoders, layers, and readouts.

Linear readout. The empirical readability reported in Section 3.4 is the AP of a ridge probe fitted on these tokens. We select the regularization strength by source-validation AP before testing.

Readout budget study. For the matched readout comparison in Section 5.1, each readout family is trained on the 3,932 source training images with two seeds and two learning rates for 8 and 16 epochs, and the linear and MLP readouts also for 24 epochs. Learning rates and epochs are selected on the 1,025 source validation images, giving 20 selected readouts per encoder. For the depth analysis in Table 4, we apply the encoder's output LayerNorm at each selected block and fit a separate ridge probe, selecting its regularization on source validation.

Probe normalization. To test whether the readability gain depends on feature scaling, we refit the probe after standardizing each feature dimension and after LDA whitening with an eigenvalue cutoff of 10^{-8}, selecting the regularization strength on source validation from a wider grid. Table 16 shows that the large gain persists. With this wider grid, the raw probe on Conv3-adapted features reaches 0.9211 AP, compared with 0.9187 for the fit reported in Section 5.

Table 16: Linear-probe AP on raw, standardized, and LDA-whitened features.

Features	Raw	Standardized	LDA
Frozen	0.5014	0.5011	0.4997
Conv3-adapted	0.9211	0.9203	0.9067
Linear-adapted	0.8953	0.8953	0.8892

Interior tokens. We also evaluate the fitted probe only on interior tokens, whose 16×16 footprint lies inside one class and whose pixels are all more than 32 pixels from the nearest pixel of the other class, keeping images with at least four tokens of each class (909 of 1,764 images). AP rises from 0.4233 for frozen features to 0.9400 for Conv3-adapted and 0.9208 for linear-adapted features.

Residual directions. Table 17 reports AP for the residual and its leading principal directions.

Table 17: Readability of the adaptation residual $\Delta\mathbf{F} = \mathbf{F}_1 - \mathbf{F}_0$ and of its leading principal directions, for the Conv3-adapted and linear-adapted encoders.

Readout input	Conv3-ad.	Linear-ad.
$\mathbf{F}_1$	0.9187	0.8953
$\Delta\mathbf{F}$ (all dims.)	0.9054	0.8705
Top-16 PCs of $\Delta\mathbf{F}$	0.9025	0.8488

Evaluation. Readouts are applied without refitting to the four CAT-Net test sets. At test time, every token on the grid is labeled edited if at least half of its pixels are edited, and authentic otherwise. Token-level AP is computed per image on the 32×32 grid, using the 1,751 of 1,764 images that contain both edited and authentic tokens; it is averaged within each dataset and then with equal weight across datasets. Intervals are paired image bootstraps within datasets.

F Paired Spatial Diagnostics

F.1 Diagnostic Protocol

Fixed checkpoints and paired inputs. We reuse the full model, linear-head and BCE-only variants, and both 448-input backbone models without retraining, retuning thresholds, or selecting new checkpoints. Each comparison contains the same 1,764 images: 920 CASIAv1, 100 Coverage, 180 Columbia, and 564 NIST16. Image identities, file hashes, and ground-truth masks are checked pairwise. Checkpoint selection is described in Appendix D; the diagnostics use the selected checkpoints, with the final-epoch comparison reported separately in Table 15. Each metric is averaged over images within a dataset, then equally across the four datasets. Metrics with an empty eligible pixel set are undefined for that image and excluded from that metric's mean.

Model-input metrics and common-grid geometry. Pixel F1, average precision (AP), precision, recall, and the near-total-miss indicator are computed at the model's input resolution: 512 for the full, linear-head, and BCE-only models, and 448 for the backbone comparison. Binary predictions use the strict threshold $p > 0.5$. Per-image F1 follows the definition used by the IMDL-BenCo evaluator. For geometric diagnostics only, model-input probabilities are bilinearly interpolated to 512×512 before thresholding; original GT masks are nearest-neighbor resized directly to that grid. Diagnostic mean F1 values for the full, linear-head, and BCE-only models are 0.847258, 0.814435, and 0.833003; the 448-input DINOv3 and DINOv2 models reach 0.833297 and 0.807262, respectively. The diagnostic scores come from reevaluating the saved checkpoints and agree within 0.1 percentage points with the original logged values retained in Tables 1 and 6.

Boundary and spatial errors. An inner contour is the foreground mask minus its 3×3 erosion, with image-exterior pixels treated as background. Boundary precision is the fraction of predicted-contour pixels within Euclidean distance t of the GT contour; boundary recall reverses the roles. Their harmonic mean is BF_t, for $t \in \{1, 2, 4\}$ pixels. We assign $\mathrm{BF}_t = 0$ when exactly one contour is empty. In particular, a nonempty GT with an empty prediction remains in the BF mean with score zero. If both contours are empty, BF is undefined and excluded. We aggregate boundary F1 across images; its precision and recall terms are used only to calculate each image's F1. Let B contain pixels within two pixels of the GT contour. Non-boundary F1 is computed on B^c, including both foreground and background. Interior FN rate is the missed fraction of GT foreground in B^c; far-background FP rate is the predicted-positive fraction of GT background in B^c. Boundary FN and FP rates use the foreground and background parts of B, respectively. Their pixel counts partition all FN and FP, but their rates have different denominators and cannot be added to allocate F1 gains. Background error rates are measured within the manipulated images.

Misses and holes. A near-total image miss has GT-foreground recall below 0.1, not necessarily an empty prediction. Enclosed-hole errors are false-negative interior pixels enclosed by the predicted foreground, using complementary four-connected background topology; hole filling is used only to measure this quantity, never to modify predictions.

Uncertainty. We resample paired image differences 2,000 times per dataset (seed 20260910), compute an equal-dataset macro-mean in each replicate, and report percentile 95% intervals.

F.2 Paired Differences

Eligible-image counts and absolute contour scores. Every metric in Table 19 is defined on all 1,764 images for each of the five configurations, and empty predictions against nonempty GT contribute zero to the boundary-F1 means. Table 18 reports absolute contour quality for these models.

Table 18: Absolute boundary F1 on the common 512×512 grid, with one-, two-, and four-pixel tolerances. Values in $[0, 1]$ are averaged over images within each dataset and equally across datasets. N is the eligible image count for each BF metric; input resolutions follow Table 6.

Configuration	$BF_1 \uparrow$	$BF_2 \uparrow$	$BF_4 \uparrow$	N
Full model	0.5261	0.6590	0.7775	1,764
Linear head	0.4829	0.6116	0.7317	1,764
BCE only	0.4509	0.5950	0.7429	1,764
DINOv3 (448)	0.4980	0.6271	0.7522	1,764
DINOv2 (448)	0.4986	0.6270	0.7419	1,764

Table 19: Paired macro-average differences in percentage points with 95% image-bootstrap intervals. Positive is better for F1, AP, precision, and recall; negative is better for error rates.

Metric	Conv3–linear	BCE+edge–BCE	DINOv3–DINOv2
F1 ↑	3.28 [2.23, 4.46]	1.43 [0.53, 2.37]	2.60 [1.37, 4.00]
AP ↑	2.76 [1.99, 3.67]	0.21 [−0.55, 1.05]	2.29 [1.04, 3.67]
$BF_1 \uparrow$	4.32 [3.53, 5.14]	7.52 [6.78, 8.24]	−0.06 [−1.03, 1.02]
$BF_2 \uparrow$	4.74 [3.84, 5.66]	6.40 [5.56, 7.28]	0.00 [−1.18, 1.18]
$BF_4 \uparrow$	4.57 [3.63, 5.66]	3.45 [2.44, 4.44]	1.04 [−0.36, 2.35]
Non-boundary F1 ↑	3.60 [2.48, 4.90]	1.04 [0.15, 2.06]	2.87 [1.49, 4.26]
Precision ↑	3.18 [2.08, 4.34]	0.07 [−0.93, 1.06]	2.90 [1.56, 4.35]
Recall ↑	2.74 [1.67, 3.94]	2.98 [2.02, 4.02]	2.25 [0.88, 3.75]
Interior FN ↓	−3.04 [−4.38, −1.85]	−2.66 [−3.89, −1.57]	−2.52 [−4.03, −1.04]
Boundary FN ↓	−2.72 [−3.79, −1.67]	−8.32 [−9.22, −7.37]	−0.91 [−2.27, 0.46]
Boundary FP ↓	0.12 [−0.76, 1.03]	2.73 [1.83, 3.66]	1.10 [0.04, 2.14]
Far-background FP ↓	−0.005 [−0.367, 0.397]	0.470 [−0.001, 0.978]	−0.902 [−1.654, −0.247]
Near-total image miss ↓	−1.64 [−3.01, −0.38]	−2.08 [−3.55, −0.74]	−3.19 [−4.75, −1.73]
Enclosed-hole FN ↓	−0.05 [−0.13, 0.04]	0.09 [−0.01, 0.23]	−0.13 [−0.25, −0.03]

G Full Post-processing Results

We report the full CASIAv1 results for Gaussian noise, Gaussian blur, and JPEG compression in Tables 21 to 23. For our models, the IMDL-BenCo evaluation code specifies Albumentations 1.3.0: noise level v specifies variance on the 0–255 pixel scale, and blur level k gives the square kernel width in pixels. JPEG level q samples integer quality from $\{q-1, q\}$. Prior-method rows reproduce the values and numbered levels in Table 2 of Mesorch's arXiv v1. Section 4.3 summarizes the main findings, and Table 20 compares clean and strongest-setting scores.

Table 20: CASIAv1 robustness at the strongest tested setting in each perturbation family: noise level 23, blur kernel size 23, and JPEG quality 50. Mesorch values are from Zhu et al. (2024) (arXiv v1, Table 2). All values are pixel-level F1; full sweeps appear in Appendix G.

Model	Clean	Noise	Blur	JPEG
Mesorch	0.8398	0.7696	0.2940	0.6552
DINOv3-L, full fine-tuning	0.8816	0.8678	0.3224	0.7716
DINOv3-L, LoRA r=32	0.9071	0.8921	0.4786	0.8060

Table 21: Robustness under Gaussian noise on CASIAv1. Pixel-level F1 at six noise levels. Prior method results from Zhu et al. (2024) (arXiv v1, Table 2). Best per column highlighted in green.

	Method	Clean	3	7	11	15	19	23	Avg.
Prior work	MVSS-Net (Dong et al., 2023)	0.5832	0.5824	0.5822	0.5764	0.5736	0.5620	0.5613	0.5744
	PSCC-Net (Liu et al., 2022)	0.6304	0.6127	0.5752	0.5540	0.5402	0.5232	0.5115	0.5639
	CAT-Net (Kwon et al., 2022)	0.8081	0.7979	0.7883	0.7832	0.7720	0.7573	0.7551	0.7802
	TruFor (Guillaro et al., 2023)	0.8208	0.7666	0.7378	0.7190	0.6947	0.6832	0.6780	0.7286
	Mesorch (Zhu et al., 2024)	0.8398	0.8205	0.8050	0.7968	0.7887	0.7780	0.7696	0.7998
ViT-S	DINOv3 + LoRA r=32	0.7870	0.7837	0.7716	0.7698	0.7584	0.7561	0.7517	0.7683
	DINOv3 + LoRA r=64	0.8025	0.7983	0.7901	0.7865	0.7752	0.7747	0.7692	0.7852
	DINOv3 + Full Fine-tuning	0.7043	0.6997	0.6946	0.6918	0.6891	0.6822	0.6789	0.6915
ViT-B	DINOv3 + LoRA r=32	0.8395	0.8297	0.8148	0.8061	0.8007	0.7962	0.7871	0.8106
	DINOv3 + LoRA r=64	0.8630	0.8493	0.8379	0.8260	0.8204	0.8137	0.8150	0.8322
	DINOv3 + Full Fine-tuning	0.8148	0.8067	0.7945	0.7834	0.7787	0.7748	0.7719	0.7893
ViT-L	DINOv3 + LoRA r=32	0.9071	**0.9027**	**0.9009**	**0.8987**	**0.8967**	**0.8889**	**0.8921**	**0.8982**
	DINOv3 + LoRA r=64	**0.9077**	0.8972	0.8957	0.8919	0.8902	0.8875	0.8868	0.8939
	DINOv3 + Full Fine-tuning	0.8816	0.8784	0.8760	0.8726	0.8698	0.8697	0.8678	0.8737

Table 22: Robustness under Gaussian blur on CASIAv1. Pixel-level F1 at six kernel sizes. Prior method results from Zhu et al. (2024) (arXiv v1, Table 2). Best per column highlighted in green.

	Method	Clean	3	7	11	15	19	23	Avg.
Prior work	MVSS-Net (Dong et al., 2023)	0.5832	0.4587	0.3097	0.2369	0.1890	0.1571	0.1392	0.2962
	PSCC-Net (Liu et al., 2022)	0.6304	0.5410	0.4531	0.3156	0.1655	0.1140	0.0775	0.3282
	CAT-Net (Kwon et al., 2022)	0.8081	0.7512	0.6532	0.5435	0.4337	0.3142	0.2142	0.5312
	TruFor (Guillaro et al., 2023)	0.8208	0.7508	0.6881	0.6032	0.4563	0.2741	0.1304	0.5320
	Mesorch (Zhu et al., 2024)	0.8398	0.7898	0.7081	0.6277	0.5328	0.4193	0.2940	0.6016
ViT-S	DINOv3 + LoRA r=32	0.7870	0.7396	0.6694	0.5818	0.4529	0.3339	0.2525	0.5453
	DINOv3 + LoRA r=64	0.8025	0.7614	0.6974	0.6116	0.4860	0.3700	0.2941	0.5747
	DINOv3 + Full Fine-tuning	0.7043	0.6453	0.5584	0.4427	0.3136	0.2338	0.1618	0.4371
ViT-B	DINOv3 + LoRA r=32	0.8395	0.8061	0.7650	0.6840	0.5455	0.3890	0.2697	0.6141
	DINOv3 + LoRA r=64	0.8630	0.8357	0.7866	0.7116	0.5285	0.3091	0.1851	0.6028
	DINOv3 + Full Fine-tuning	0.8148	0.7655	0.7112	0.6222	0.4947	0.3650	0.2632	0.5767
ViT-L	DINOv3 + LoRA r=32	0.9071	**0.8857**	**0.8476**	**0.8069**	**0.7218**	**0.5900**	**0.4786**	**0.7482**
	DINOv3 + LoRA r=64	**0.9077**	0.8798	0.8371	0.7941	0.6670	0.4810	0.3556	0.7032
	DINOv3 + Full Fine-tuning	0.8816	0.8545	0.8176	0.7543	0.6284	0.4518	0.3224	0.6729

Table 23: JPEG robustness on CASIAv1 at six quality settings. Prior results are from arXiv v1, Table 2 of Zhu et al. (2024); column maxima are green.

	Method	Clean	100	90	80	70	60	50	Avg.
Prior work	MVSS-Net (Dong et al., 2023)	0.5832	0.5695	0.5446	0.5170	0.4906	0.4489	0.3888	0.5061
	PSCC-Net (Liu et al., 2022)	0.6304	0.6220	0.5789	0.4930	0.4518	0.3846	0.2869	0.4925
	CAT-Net (Kwon et al., 2022)	0.8081	0.7896	0.7858	0.7431	0.7230	0.6838	0.6132	0.7352
	TruFor (Guillaro et al., 2023)	0.8208	0.8060	0.7938	0.7017	0.6852	0.6329	0.4942	0.7049
	Mesorch (Zhu et al., 2024)	0.8398	0.8312	0.8194	0.7716	0.7706	0.7285	0.6552	0.7738
ViT-S	DINOv3 + LoRA r=32	0.7870	0.7772	0.7645	0.7211	0.7077	0.6693	0.5817	0.7155
	DINOv3 + LoRA r=64	0.8025	0.7954	0.7818	0.7464	0.7286	0.6861	0.5981	0.7341
	DINOv3 + Full Fine-tuning	0.7043	0.6924	0.6777	0.6287	0.6033	0.5598	0.4855	0.6217
ViT-B	DINOv3 + LoRA r=32	0.8395	0.8316	0.8186	0.7846	0.7659	0.7355	0.6577	0.7762
	DINOv3 + LoRA r=64	0.8630	0.8548	0.8463	0.8117	0.7982	0.7843	0.7103	0.8098
	DINOv3 + Full Fine-tuning	0.8148	0.8059	0.7943	0.7570	0.7429	0.7106	0.6292	0.7507
ViT-L	DINOv3 + LoRA r=32	0.9071	**0.9058**	0.8946	0.8755	0.8753	0.8598	**0.8060**	0.8749
	DINOv3 + LoRA r=64	**0.9077**	0.9035	**0.8959**	**0.8775**	**0.8758**	**0.8622**	0.8046	**0.8753**
	DINOv3 + Full Fine-tuning	0.8816	0.8778	0.8749	0.8485	0.8459	0.8312	0.7716	0.8474

H CONTROLLED COPY-MOVE EXPERIMENT

To separate editing role from pixel content, we construct a controlled copy-move set in which the pixels of edited regions are fixed exactly. For each of 100 authentic COVERAGE photographs, we copy a 128×128 square region pixel for pixel to a second location, without rescaling, blending, or recompression, and repeat the copy in the opposite direction, producing 200 edited images. The source and pasted-target interiors therefore contain identical pixel content. Within each image, we compare all pairs of tokens drawn from the central 4×4 token grids of the target and source regions and compute the role AUC

$$\mathrm{AUC}_{\mathrm{role}} = \Pr[s(x_t) > s(x_s)] + \tfrac{1}{2} \Pr[s(x_t) = s(x_s)] \,, \tag{3}$$

where s is the linear probe of Section 3.4, and x_t and x_s are target and source tokens, respectively. We average the two copy directions within each photograph and then average across photographs. The probe ranks the target above the source with a role AUC of 0.7029 on frozen features, 0.9879 on Conv3-adapted features, and 0.9663 on linear-adapted features; the 16 leading residual directions of Section 5.3 give 0.9862 and 0.9569. The frozen encoder therefore already separates the two roles to some extent; adaptation sharpens this distinction substantially. The comparison holds the source and target interior pixels fixed.