MATCH EVERY PATCH ONCE: A LOGICAL-STRUCTURAL TRADEOFF IN ANOMALY SCORING

Anonymous authors
Paper under double-blind review

ABSTRACT

Normal-only image anomaly detection often scores an image by matching each test patch independently to its closest patch in a normal reference. This allows many test patches to match the same reference patch, making changes in the number or composition of otherwise normal parts difficult to detect. We instead require a one-to-one patch assignment between a test image and a complete normal reference, keeping the features, patch costs and aggregation fixed. We define the assignment excess as the additional cost introduced by one-to-one matching and show that it exactly accounts for the resulting changes in anomaly ranking. The constraint improves sensitivity to logical anomalies but can reduce sensitivity to structural defects, a tradeoff that depends strongly on how normal references are constructed. Used as an auxiliary signal in a trained EfficientAD-M detector, a calibrated form of the assignment excess improves logical anomaly detection in all five training runs. Patch correspondence and reference construction must therefore be specified together in anomaly scoring.

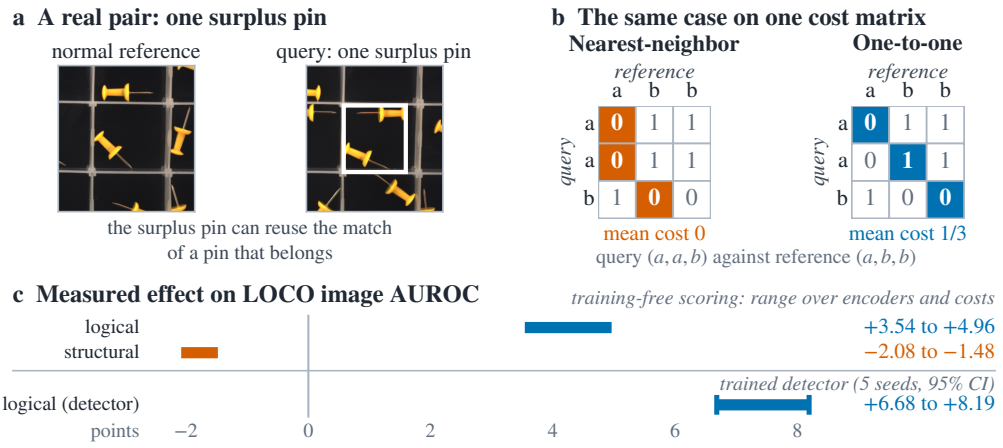

Figure 1: **A surplus pin rides free under reuse; matching once charges for it. (a)** Each cell should hold one pin. A LOCO pushpins query with one surplus pin (box: ground-truth region) beside a complete normal reference. Nearest-neighbor matching can send the surplus pin to the reference patch of a pin that belongs, at almost no cost. **(b)** Query (a, a, b) against reference (a, b, b), cost zero within a type and one across types. Reuse costs 0; matching every patch exactly once must send one a to a b and pays $1/3$. **(c)** Change in LOCO image AUROC. Rows one and two span both encoders on the original references (Table 5); row three is the 95% interval over five EfficientAD-M seeds with conditional excess fused (Table 4).

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.

1 INTRODUCTION

A patch-based anomaly score sees what each part of an image looks like. It does not see how many parts there are. Industrial anomaly detectors are trained on normal images alone and must catch two kinds of defect at test time. Structural anomalies are local, visible in one patch. Logical anomalies change the number, composition or arrangement of parts, such as a missing screw or a duplicated pin in an image whose patches look normal on their own (Bergmann et al., 2022). Scores that compare each test patch with its nearest normal patch, as PatchCore and AnomalyDINO do (Roth et al., 2022; Damm et al., 2025), are built for the first kind. Nothing stops many test patches from reusing the same reference patch. Under discrete patch types, reuse discards how many times each shared type occurs. A surplus or missing part stays cheap while its patches keep close normal matches. A patch-matching detector decides, usually implicitly, whether reuse is free and what the reference is.

We forbid that reuse and measure what it discards. Each patch of the test image must match a patch of its own in a single complete normal image, through a linear assignment on the same patch-cost matrix. A surplus part now has nowhere to go: the reference patch it would have borrowed is taken, so the assignment must pay a cross-type cost somewhere (Figure 1). We call this extra cost the *assignment excess*. The one-to-one score is exactly the nearest-neighbor score plus the excess, so every ranking change between the two rules is a difference in excess (Section 3.4). The constraint reads only the multiset of patches. A changed count or composition raises its cost. A rearrangement that leaves the multiset unchanged does not. We measure the two subtypes separately (Appendix A).

On MVTec Logical Constraints (MVTec LOCO) (Bergmann et al., 2022), forbidding reuse raises logical image area under the receiver operating characteristic curve (AUROC) by 3.54 to 4.96 points. Structural AUROC falls by 1.48 to 2.08 points (Figure 2). Within each encoder the two rules share everything else: frozen DINOv2 (Oquab et al., 2024) or DINOv3 (Siméoni et al., 2025) features, the normal reference, the patch costs and the averaging.

The matching rule and the reference must be specified together. Merge the patches of many normal images into one pooled bank and the surplus part finds unmatched capacity, so the contrast nearly disappears. Normal images can pay excess too, which is why a larger matching score need not improve ranking (Section 3.4).

Contributions.

- *Reuse costs nothing in the score.* Under discrete patch types shared with the reference, nearest-neighbor matching ignores how often each type occurs, and one-to-one assignment does not (Proposition 1). On one cost matrix, the excess, zero exactly under Hall's condition, accounts for every ranking change within each category (Section 3).
- *When counts matter.* Forbidding reuse raises sensitivity to count and composition anomalies and lowers structural sensitivity. The count gain recurs on CAD-SD (Co-occurrence Anomaly Detection Screw Dataset), DigitAnatomy and VID-AD (Vision-Induced Distraction) and reverses on VisA under DINOv3. On LOCO, pooling nearly removes it and per-reference scoring restores it (Sections 4.2–4.4).
- *Excess in a detector.* In EfficientAD-M, conditional excess raises logical AUROC in all five seeds within a one-point structural harm margin; PatchCore pays beyond it (Section 4.4).

2 RELATED WORK

Detecting anomalies from normal data alone. Normal-only anomaly detection must identify defects absent from training (Bergmann et al., 2019; Zou et al., 2022). Embedding methods compare features with a stored normal description (Defard et al., 2021; Roth et al., 2022; Gudovskiy et al., 2022), distillation methods use the residual between a student and frozen teacher (Bergmann et al., 2020; Deng & Li, 2022; Batzner et al., 2024), and synthesis methods create defects to train a discriminative model (Li et al., 2021; Zavrtanik et al., 2021; Liu et al., 2023b). Dinomaly reconstructs features from a frozen DINO backbone in class-separated or multi-class training (Guo et al., 2025), while MuSc mutually scores unlabelled test images and reuses a patch across many comparisons (Li et al., 2024). These approaches target locally visible defects. Logical anomalies violate that premise because each patch in a logically wrong image can itself be normal (Bergmann et al., 2022).

Representing composition. Composition-aware methods represent the image without imposing the correspondence tested here. ComAD and PSAD summarize segmented components (Liu et al., 2023a; Kim et al., 2024), and SINBAD summarizes projected feature distributions (Cohen et al., 2024); an extra part changes the feature distribution without requiring a decision about its matching reference patch. LogiCo learns component- and patch-level reconstruction with a segmentation-map discriminator (Zhang et al., 2026), SALAD models the distribution of object composition maps directly and extracts those maps without hand-made component labels (Fučka et al., 2025), while Li et al. (2026) combine five frozen appearance and counting branches through normal-set calibration and soft-OR fusion.

Matching normal structure. Recent systems match normal structure while changing more than the correspondence constraint. LogSAD runs a reusable patch branch beside a Hungarian branch that matches Segment Anything regions against a pooled bank, so the two branches differ in unit, features and segmentation as well as in matching rule (Zhang et al., 2025). SAM-LAD partially assigns segmented objects for zero-shot logical anomaly detection (Peng et al., 2025). We change correspondence alone, with no segmentation. ObjectCore couples padded bipartite matching with support retrieval, object descriptors, and a structural branch (Fučka et al., 2026). HETMM adds a template-to-query nearest-neighbor pass that catches missing objects on MVTec LOCO while still allowing reuse (Chen et al., 2025). On our costs, coverage without exclusivity buys part of the logical gain at a larger structural cost (Appendix A.3). Local and global prototypes are learned with optimal transport on MVTec AD and MVTec LOCO (Trombetta & Lartizien, 2026), and UniVAD v2 learns support-conditioned transport and fusion (Gu et al., 2026).

Assignment and transport as matching rules. One-to-one assignment (Kuhn, 1955) has served other jobs: comparing shapes through point correspondences (Belongie et al., 2002) and freeing DETR from non-maximum suppression (Carion et al., 2020). Chamfer-style set distances allow reuse (Fan et al., 2017). As a point-cloud completion metric, density-aware Chamfer distance penalizes a match by how many points share it, calling one-to-one mapping over harsh (Wu et al., 2021). The Hungarian method solves such assignments exactly (Kuhn, 1955), entropic relaxations make transport tractable at scale (Cuturi, 2013; Peyré & Cuturi, 2019), and an earth mover's distance between local feature sets classifies images from few examples (Zhang et al., 2020). In recent logical-anomaly systems, matching or transport arrives together with segmentation, object detectors, learned prototypes or a trained head (Fučka et al., 2026; Trombetta & Lartizien, 2026; Gu et al., 2026; Zhang et al., 2025); Table 19 lists which of these vary the matching rule alone. Our matcher has no parameters of its own, so only the constraint moves the score; a relaxed matcher also moves it through its own parameters (Appendix F).

From local evidence to an image score. Image-level scores also differ in how they aggregate patch evidence, through a reweighted patch maximum over a coreset memory (Roth et al., 2022) or calibrated map descriptors (Chae et al., 2026). Our nearest-neighbor mean keeps PatchCore's matching step but averages over one complete reference, without coreset or reweighting. The detector experiments hold the baseline and calibration fixed and vary only the added statistic (Appendix C.1).

3 Method

Both matching rules and their controls run on one patch-cost matrix. Table 1 reports every rule on DINOv2 raw costs with encoder-native references.

3.1 Features and Costs

Let $\mathbf{Q} = (\mathbf{q}_1, \ldots, \mathbf{q}_N)$ and $\mathbf{R} = (\mathbf{r}_1, \ldots, \mathbf{r}_N)$ be ordered query and reference patch-feature sequences. Bold lowercase symbols are vectors, bold uppercase sequences, and light uppercase matrices or counts. K denotes the number of complete reference images pooled into one bank, and R the number of complete references scored one at a time. We use squared-Euclidean costs and also evaluate pairwise median normalization:

$$D_{ij} = \|\mathbf{q}_i - \mathbf{r}_j\|_2^2, \qquad C_{ij} = D_{ij} / \max\{\mathrm{median}_{ab}\, D_{ab}, 10^{-8}\}. \tag{1}$$

Every rule receives the same realized matrix, either D or C. Below, C denotes the selected matrix. Pairwise normalization can change image rankings because its scale varies across image pairs. No coordinates enter; any position or context information is the encoder's own.

3.2 Nearest-Neighbor Matching and Assignment

The nearest-neighbor (NN) mean, also called the reusable or directed mean, matches each query patch to its cheapest reference patch, independently of the other query patches. The assignment mean minimizes the mean cost over all one-to-one matchings, that is, over all permutations:

$$s_{\text{reuse}} = \frac{1}{N}\sum_i \min_j C_{ij}, \qquad s_{\text{bij}} = \frac{1}{N}\min_{\pi\in\mathcal{S}_N}\sum_i C_{i,\pi(i)}. \tag{2}$$

One-to-one assignment uses every reference patch exactly once. It therefore forbids reuse and requires complete coverage at the same time, so $s_{\text{bij}} \geq s_{\text{reuse}}$ for every image. Because the permutations are the vertices of the doubly stochastic matrices, s_{bij} is also the balanced optimal-transport cost of the pair at zero entropy (Appendix F). The optimal mean does not depend on which optimal assignment is chosen, and it is invariant to reordering rows or columns.

3.3 Coverage Controls, and When Assignment Incurs Cost

The coverage controls ask a different question: whether the query also covers the reference. With $u_i = \min_j C_{ij}$ and $v_j = \min_i C_{ij}$, they are

$$s_{\text{C}} = \frac{1}{2N}\left(\sum_i u_i + \sum_j v_j\right), \qquad s_{\text{H}} = \max\{\max_i u_i, \max_j v_j\}. \tag{3}$$

These are the Chamfer mean and the Hausdorff maximum. Three maximum-based rules remain: the directed maximum $\max_i u_i$, the largest edge of a minimum-sum assignment, and a partial-assignment maximum. Each control changes coverage direction, aggregation, or both, which separates those choices from the one-to-one constraint (exact definitions in Appendix F.2). At equal cardinality the one-to-one rule bundles both, and Appendix A.3 pulls them apart.

Consider the toy pair of Figure 1: query (a, a, b) and reference (a, b, b), with zero within-type cost and unit cross-type cost. Each query type occurs in the reference, so every directed or bidirectional coverage score is zero. A one-to-one matching must cross types once and has mean cost $1/3$.

Proposition 1 (What reuse discards under discrete types)**.** *Suppose features take finitely many types, within-type cost is zero, cross-type cost is at least $\delta > 0$, and the query and reference contain the same set of types. Then every nearest-neighbor and coverage score is zero. Write n_t and m_t for the query and reference counts of type t. If the counts differ, at least $\frac{1}{2}\sum_t |n_t - m_t|$ matches must cross types, so the assignment mean is at least $\delta \sum_t |n_t - m_t|/(2N)$.*

The premises are ideal: contextual features may blur the types, and a pooled bank breaks the counts (proof in Appendix E.1); Section 4.3 measures the second.

A continuous separation under a fixed reference. A continuous companion states when one-to-one matching must pay while nearest-neighbor matching need not, on the same squared-Euclidean costs.

Proposition 2 (Substitution separation under a fixed reference)**.** *Let a query be formed from a reference $\mathbf{R} = (r_1, \ldots, r_N)$ by substituting token r_a with r'_b satisfying $\|r'_b - r_b\| \leq \delta$ for some other reference token r_b, and let $\gamma_a = \min_{j\neq a}\|r_a - r_j\|$. If $0 \leq \delta < \gamma_a$, every directed reusable per-token score is at most δ^2 (each query token keeps a close match), so the reusable mean is at most δ^2/N, while any mass-preserving matching rule whose uniform marginals force every reference token to receive mass pays at least $(\gamma_a - \delta)^2$ on the displaced token's column, so its mean cost is at least $(\gamma_a - \delta)^2/N$. The one-to-one assignment mean obeys the same bound. As $\delta \to 0$ the ratio of the matching lower bound to the reusable upper bound diverges. The bound uses only marginal feasibility and nonnegative costs, never optimality. Without the hypothesis $\delta < \gamma_a$ the bound reads $\max(\gamma_a - \delta, 0)^2/N$.*

Table 1: **All else fixed, forbidding reuse raises logical and lowers structural AUROC.** LOCO category-macro image AUROC (%), DINOv2, raw costs, encoder-native references; Combined pools both strata against shared normals. The first two rows differ only in the matching rule; the other five also change coverage or aggregation. Shading marks the rule under test.

Score	Logical ↑	Structural ↑	Combined ↑
NN mean (reuse allowed)	73.62	69.75	71.55
Assignment mean (one-to-one)	78.30	67.93	73.22
Directed maximum	63.83	69.65	66.19
Min-sum assignment: max	71.13	62.26	67.00
Partial assignment (95%)	76.17	66.51	71.40
Chamfer mean	75.43	68.11	71.75
Hausdorff maximum	68.10	67.98	67.82

Proposition 3 (Retrieved banks)**.** *If each retrieved reference* $\mathbf{R}$ *is scored by its own one-to-one map with the query patches* $\mathbf{q}_i$*, and every such* $\mathbf{R}$ *satisfies* $\gamma(\mathbf{Q}, \mathbf{R}) = \max_j \min_i \|\mathbf{q}_i - r_j\| \geq m > 0$*, then the mean matching cost is at least* m^2/N*. A retrieved reference that carries the anomalous token composition can make the bound vacuous.*

The count bound is symmetric in the query and reference counts, so under the same premises an incomplete reference makes normal images pay excess too, and the contrast shrinks. Appendix B.2 measures both failure modes: a reference that is itself missing a part cuts the logical contrast from $+4.68$ to $+0.96$, and contaminating the bank with anomalous images lowers it monotonically to $+2.55$ $[+0.99, +4.13]$ at half contamination.

3.4 Assignment Excess Accounts for Every Ranking Change

Assignment can help one anomaly stratum and hurt another because it raises normal and anomalous scores by different amounts. The amount it adds is the *assignment excess*, the extra cost one-to-one assignment pays over nearest-neighbor matching:

$$e = s_{\text{bij}} - s_{\text{reuse}} \geq 0. \tag{4}$$

For an anomalous image a and a normal image n, the one-to-one score margin equals the nearest-neighbor margin plus $e_a - e_n$. The constraint corrects a pairwise ordering when $e_a - e_n$ outweighs a negative nearest-neighbor margin, and reverses it when $e_n - e_a$ outweighs a positive one. A larger excess alone only shifts the margin. The excess is zero exactly when the nearest-neighbor choices, counting ties, already contain a one-to-one matching (Hall's condition; Appendix E.2). Only then does forbidding reuse cost a nearest-neighbor score nothing. Every observed query–reference pair has positive excess, so ranking changes come from how its size differs between images. Counting corrected and reversed normal–anomaly pairs, with ties at half credit, recovers the AUROC change. Because pairs share images, we assess uncertainty at the image level.

3.5 Assignment Excess as a Detector Signal

We keep a trained detector's baseline score, calibration and fusion weight fixed, and add one matching statistic at unit weight. It is either the assignment excess or the conditional excess, which subtracts the median excess of the nearest fitting normals in reusable score; the fusion keeps only the positive part of this statistic after scaling. The detector is not retrained (Section 4.4).

4 Experiments

4.1 Setup

Images and representations. LOCO provides 1,568 test images in five categories: 575 normal, 561 logical-anomalous, and 432 structural-anomalous. We compare each anomaly stratum with normal images from the same category. Inputs are 224×224 for DINOv2 (Oquab et al., 2024) and 256×256 for DINOv3 (Siméoni et al., 2025). We bilinearly resize RGB images to

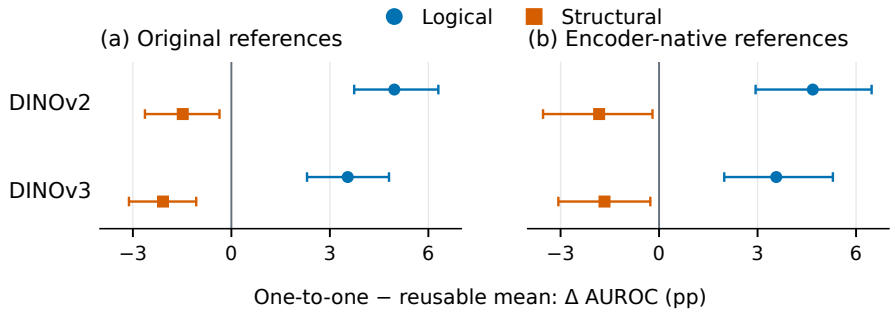

Figure 2: **The tradeoff holds on original and encoder-native references.** One-to-one minus reusable mean, LOCO category-macro AUROC points, raw costs, same test images: (a) original and (b) encoder-native references, intervals from Table 5. Every logical interval lies above zero and every structural one below.

a square with antialiasing, convert them to tensors, and normalize them with ImageNet channel means $(0.485, 0.456, 0.406)$ and standard deviations $(0.229, 0.224, 0.225)$. Each encoder supplies 256 final-layer normalized patch features of dimension 1,024; we remove class and register tokens and train nothing. We treat each encoder, at its own input resolution, as one representation.

Normal references. For the original comparison, both encoders use the same 1,568 query–reference pairs, each reference selected without labels by mean-pooled feature retrieval. We call the stored set of complete normal images a *reference bank*. Unlike a PatchCore memory bank, it keeps each image's complete patch grid. In a second comparison, *encoder-native retrieval*, each encoder selects its own reference: the most similar normal image of the same category among the 1,778 training images. This changes 1,497 DINOv2 and 339 DINOv3 references and leaves the test queries unchanged. Each query is scored against a complete normal image of its category, which assumes correct routing, here given by LOCO's category labels; Appendix B.2 tests incomplete and contaminated banks.

Evaluation. We report category-macro image AUROC, averaging categories with equal weight. Intervals are paired bootstrap intervals over test images, resampled within each category and anomaly stratum (Appendix G). We report all seven rules for both encoders and both normalizations and chose none on test data. We fixed the single-reference primary contrast before computing new features; it was chosen after an earlier study on the same pairs. We fixed the later encoder-native family before retrieving or scoring its references.

4.2 Logical Gains and Structural Costs

On a single complete reference, the one-to-one rule raises logical AUROC and lowers structural AUROC for both encoders and both cost normalizations (Figure 2). On the original references, the logical gain is 3.54 to 4.96 points and the structural loss 1.48 to 2.08 points (Table 5). All eight pointwise intervals, across two encoders, two cost normalizations and two strata, lie on their side of zero, and the two normalizations agree to within 0.08 points. Encoder-native references keep the direction of all four raw-cost contrasts (Figure 2).

On a fixed patch grid, a rearrangement that leaves the encoded patch multiset unchanged leaves the assignment cost unchanged (Section 3). An exploratory split by LOCO's defect subtypes is consistent with this. Anomalies that change the count or composition of parts give 3.75 to 5.29 points across the four configurations. The 59 images whose anomaly only rearranges otherwise correct parts give -0.48 to -0.04, every interval covering zero (Appendix A). Used alone as a score, the excess nearly matches the assignment mean on DINOv2 logical anomalies and is weak on structural ones: category-macro AUROC 77.45 logical and 57.00 structural for DINOv2 raw costs, and 78.58 and 58.69 for DINOv3 (Appendix A).

The median per-category gain is positive in all eight configuration-by-protocol cells ($+1.23$ to $+3.66$). Breakfast box supplies the largest share for both encoders; without it, all four encoder-native logical configurations stay above zero at $+1.13$ to $+3.07$, and on the original references

Table 2: **The one-to-one mean leads both logical columns.** LOCO category-macro image AUROC (%), original references, raw costs. All seven rules share features and costs within an encoder; only the first two rows differ in the matching rule alone. Maximum-based controls lead the structural columns.

	DINOv2		DINOv3	
Score	Logical ↑	Structural ↑	Logical ↑	Structural ↑
NN mean (reuse allowed)	72.98	69.73	77.41	73.31
Assignment mean (one-to-one)	77.95	68.25	80.96	71.23
Chamfer mean	74.54	68.05	79.35	72.30
Directed maximum	63.27	69.89	68.60	77.53
Min-sum assignment: max	71.61	61.09	77.78	67.05
Partial-assignment maximum	76.64	66.00	77.96	66.26
Hausdorff maximum	69.09	68.41	78.00	77.82

only the DINOv3 pair-median interval reaches zero. The structural loss survives the same omission (Appendix B.3).

4.3 Coverage, Aggregation, and Reference Construction

Coverage and aggregation. Among the maximum-based scores in Table 2, the largest edge of a minimum-sum assignment falls below the Chamfer mean on both anomaly types in all four configurations, and below the Hausdorff maximum in both DINOv3 configurations. Among the means, the order changes: the assignment mean exceeds the Chamfer mean in both DINOv2 strata, and for DINOv3 it is higher on logical anomalies but lower on structural ones. The maximum-edge score and the assignment mean can therefore point in opposite directions. Pair-median costs keep the main tradeoff direction.

Reference construction. How the reference is built changes normal support and also assignment capacity, the number of reference patches the assignment may choose among. We keep every LOCO query and retrieve nested top-one and top-five reference sets. Pooling the patches of K complete reference grids into one *pooled bank* gives assignment surplus columns. A separate per-reference construction scores each rule against each complete reference on its own and keeps that rule's minimum over the references.

With five pooled references the surplus part finds spare columns: the logical contrast nearly disappears for both encoders, and the logical and structural intervals reach zero. Per-reference scoring restores it, with pointwise intervals and each rule free to pick its own reference. The bounds of Section 3 do not cover a pooled bank. Coverage alone, with reuse still allowed, recovers about half of the logical gain at a larger structural cost (Appendix A.3). Allowing a second use per reference patch, with the bank fixed, keeps only about six percent of it (Table 8); this step also drops the requirement that every reference patch be matched. Pooling instead adds spare reference patches and changes normal support. In a sensitivity record over 120 draws of complete references, the mean logical contrast stays positive for both encoders (Appendix F.4).

Per-reference banks. A per-reference bank draws distinct complete normal references per category, with $R \in \{1, 2, 4, 8\}$. Pooled NN receives the concatenated $R \times 256$ pooled bank. Assignment instead solves one 256×256 assignment against each selected grid (Appendix F.3 gives its cost) and averages those assignment means, so its capacity is preserved. Both arms therefore use the same selected support, and averaging over references is the only additional reduction on the assignment side. With 2,000 paired image-bootstrap contrasts, the per-image mean assignment has positive logical intervals at $R = 2$ and $R = 4$ for both encoders. The DINOv3 interval at $R = 8$ includes zero, and the structural intervals show costs in several cells (Table 3).

Same-reduction check. When both arms reduce over references in the same way, all twelve logical contrasts at $R \in \{2, 4, 8\}$ stay positive with pointwise intervals above zero, the smallest $+4.42$

Table 3: **Scoring references one at a time keeps the logical gain.** Assignment minus pooled NN, LOCO equal-category macro AUROC points, banks of R complete references. *excl. 0*: the 95% interval excludes zero (Table 10(b)). The gain holds to four references on both encoders; for DINOv3 at eight it reaches zero and the structural cost peaks.

Encoder	R	Logical	excl. 0	Structural	excl. 0
DINOv2	1	$+6.13$	yes	-0.37	no
	2	$+7.25$	yes	-0.79	no
	4	$+7.22$	yes	-0.67	no
	8	$+5.77$	yes	-2.39	yes
DINOv3	1	$+5.58$	yes	-1.68	yes
	2	$+5.63$	yes	-2.68	yes
	4	$+5.02$	yes	-4.94	yes
	8	$+0.55$	no	-6.84	yes

Table 4: **Fused into EfficientAD-M, conditional excess raises logical AUROC in every training seed.** Top: LOCO equal-category macro image AUROC (%) of the seed-0 detector (library-trained, App. C.2), each matching statistic added at unit weight; +Excess is the assignment excess and +Cond. the conditional excess (Section 3.5). Bottom: +Cond. minus baseline in AUROC points for five training seeds, with 95% t-intervals over seeds; both lie above zero.

Stratum	Base	+Reuse	+Chamfer	+Excess	+Cond.
Logical	76.56	80.52	81.05	84.20	84.26
Structural	87.69	88.74	88.78	88.42	88.40

Δ AUROC	s0	s1	s2	s3	s4	Interval
Logical	$+7.70$	$+8.05$	$+7.71$	$+6.49$	$+7.23$	$[+6.68, +8.19]$
Structural	$+0.71$	$+0.93$	-0.04	$+0.37$	$+0.65$	$[+0.06, +0.99]$

points (Table 10c). Measured the same way, the structural endpoint shows the cost: DINOv2 intervals cross zero and five of six DINOv3 cells lie below it (Table 10).

4.4 Beyond LOCO and Inside a Detector

Count anomalies beyond LOCO. The count-anomaly gain is positive on CAD-SD (Ishida et al., 2023; CAD-SD repository, 2026), DigitAnatomy (Xiang et al., 2023; SQUID repository, 2026) and VID-AD (Nakata et al., 2026). On VID-AD quantity anomalies the rule gains $+10.64$ points over the reusable mean and $+9.98$ over Chamfer, with corrected lower bounds $+7.14$ and $+7.04$. DigitAnatomy, a synthetic missing-component population, gives $+6.46$ and $+22.60$ over the reusable mean on DINOv2/WideResNet-50 (WRN50). On VisA (Zou et al., 2022) count anomalies (Table 18) the rule gains $+1.87$ for DINOv2 and loses 0.84 for DINOv3, consistent with Section 3.4 and Table 9: there DINOv3's excess is smaller and less discriminative, and its baseline is already stronger. With every anomaly type over all twelve categories, the rule loses 0.74 for DINOv2, whose interval reaches zero, and 2.29 for DINOv3. On Real-IAD (Wang et al., 2024) the structural sign follows the logical sign in 16 of 18 cells, two short of the all-cells criterion; the two exceptions share one arm and bank size, and their intervals span zero (Appendix D.2).

Inside a detector. We add the signal to two detectors not designed around it, as in Section 3.5. On EfficientAD-M, the medium EfficientAD with student–teacher and autoencoder branches, conditional excess raises logical macro AUROC by 7.7 points over the baseline, and by 3.7 and 3.2 over the reusable and Chamfer additions, while structural AUROC moves by $+0.7$ and clears the harm margin, the -1 point structural loss we accept (Table 14). All arms share one EfficientAD-M that we trained, so only the added statistic differs. Across five training seeds, fused conditional excess gives logical gains of 6.49 to 8.05 points and structural changes between -0.04 and $+0.93$, and both intervals lie above zero (Table 4). Omitting each category in turn keeps every logical interval above zero; without breakfast box the structural change is -0.65, inside the harm margin (Table 15). Simple excess gives nearly the same logical AUROC (Table 14). On PatchCore the tradeoff of Section 4.2 reappears: conditional excess raises logical AUROC and costs 1.4 structural points, beyond its -1 harm margin (Appendix C.1). The recipe and the bounds are in Appendix C.2.

5 CONCLUSION

When patch types are discrete and shared with the reference, a nearest-neighbor score sees what each part looks like but not how many there are. Matching each test patch to its own patch in one complete normal image fixes that. A missing or extra part now costs something. The assignment excess accounts for the whole ranking change, within each category. The rule wins when that excess beats the nearest-neighbor margin. It does on CAD-SD, DigitAnatomy and VID-AD. It loses on VisA under DINOv3, where the excess is smaller and the baseline stronger. Local defects pay for it in structural AUROC. Rearranging correct parts changes no count. It gains nothing measurable in an exploratory split.

The reference is part of the scoring rule. A pooled bank gives a surplus part spare columns. A complete reference does not. Pooling five references shrinks the gain to near zero; scoring the same five whole, one at a time, brings it back.

In practice: for a missing or surplus part, score each test image against one complete normal image at a time. Forbid reuse. Check the gain on your encoder first. For local defects, keep the nearest-neighbor score, it holds more structural AUROC. The excess also bolts onto a trained patch detector with no retraining, as it does here with EfficientAD-M.

REPRODUCIBILITY STATEMENT

The appendices give the feature extraction, retrieval, cost, reduction and calibration protocols, and the exact image counts and splits. Every score-level contrast can be recomputed from fixed per-image scores, 1,568 rows for LOCO and 2,162 for VisA; each row holds every scoring rule for both encoders under both cost normalizations. Appendix G lists which analyses were fixed in advance and which came later. Retraining the seed-0 detector from scratch gives 83.56 logical macro AUROC against the original 84.26 (Appendix C.3). The benchmarks are third-party and remain under their own licences.

AI USE STATEMENT

In this work, we used generative AI tools (large language models) to propose and refine hypotheses, to design and provide feedback on the experiments, to implement methods and write analysis code, to clean and reformat the cached benchmark data, to assist in the writing and checking of proofs, and to interpret results. We have not used generative AI tools to generate synthetic data sets, and translation and qualitative data analysis are not applicable to this work. Additionally, we used generative AI tools for brainstorming, to identify relevant literature, to create and edit software code and figures, to draft parts of the paper and edit it to improve readability, and to format references. We reviewed all AI-assisted work. Every reported number comes from a recorded experiment output; we checked the code and the derivations in Appendix E, and that each reference exists and matches its bibliographic record. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work uses public industrial inspection datasets: MVTec LOCO AD (CC BY-NC-SA 4.0) and VisA (CC BY 4.0) under their licenses, and the others under their original access terms. The images show manufactured objects; no human subjects or personal data are involved. The intended use is quality inspection.

REFERENCES

Samet Akcay, Dick Ameln, Ashwin Vaidya, Barath Lakshmanan, Nilesh Ahuja, and Utku Genc. Anomalib: A deep learning library for anomaly detection. In *IEEE International Conference on Image Processing (ICIP)*, pp. 1706–1710, 2022. doi: 10.1109/ICIP46576.2022.9897283.

Kilian Batzner, Lars Heckler, and Rebecca König. EfficientAD: Accurate visual anomaly detection at millisecond-level latencies. In *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, pp. 128–138, 2024.

Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(4):509–522, 2002. doi: 10.1109/34.993558.

Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. MVTec AD — a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

Paul Bergmann, Kilian Batzner, Michael Fauser, David Sattlegger, and Carsten Steger. Beyond dents and scratches: Logical constraints in unsupervised anomaly detection and localization. *Int. J. Comput. Vis.*, 130(4):947–969, 2022. doi: 10.1007/s11263-022-01578-9.

CAD-SD repository. Co-occurrence Anomaly Detection Screw Dataset (CAD-SD). Official dataset repository, 2026. URL https://github.com/IshidaKengo/Co-occurrence-Anomaly-Detection-Screw-Dataset. Repository documentation accessed 6 September 2026; access year, not dataset publication year.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proc. European Conf. Comput. Vis. (ECCV)*, volume 12346 of *Lecture Notes in Computer Science*, pp. 213–229. Springer, 2020. doi: 10.1007/978-3-030-58452-8_13.

Joongwon Chae, Lihui Luo, Yang Liu, Runming Wang, Dongmei Yu, Zeming Liang, Xi Yuan, Dayan Zhang, Zhenglin Chen, Peiwu Qin, and Ilmoon Chae. StructCore: Structure-aware image-level scoring for training-free unsupervised anomaly detection. *arXiv preprint arXiv:2602.17048*, 2026. URL https://arxiv.org/abs/2602.17048v2.

Zixuan Chen, Xiaohua Xie, Lingxiao Yang, and Jian-Huang Lai. Hard-normal example-aware template mutual matching for industrial anomaly detection. *Int. J. Comput. Vis.*, 133(5):2927–2949, 2025. doi: 10.1007/s11263-024-02323-0.

Niv Cohen, Issar Tzachor, and Yedid Hoshen. Set features for anomaly detection. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=aukOnn7J4M.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 26, 2013. URL https://proceedings.neurips.cc/paper/2013/hash/af21d0c97db2e27e13572cbf59eb343d-Abstract.html.

Simon Damm, Mike Laszkiewicz, Johannes Lederer, and Asja Fischer. AnomalyDINO: Boosting patch-based few-shot anomaly detection with DINOv2. In *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, pp. 1319–1329, 2025. doi: 10.1109/WACV61041.2025.00136.

Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. PaDiM: A patch distribution modeling framework for anomaly detection and localization. In *International Conference on Pattern Recognition Workshops*, 2021.

Hanqiu Deng and Xingyu Li. Anomaly detection via reverse distillation from one-class embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Haoqiang Fan, Hao Su, and Leonidas J. Guibas. A point set generation network for 3D object reconstruction from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Matic Fučka, Vitjan Zavrtanik, and Danijel Skočaj. SALAD: Semantics-aware logical anomaly detection. In *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pp. 21843–21852, 2025. doi: 10.1109/ICCV51701.2025.02028.

Matic Fučka, Vitjan Zavrtanik, and Danijel Skočaj. ObjectCore – efficient few-shot logical anomaly detection using object representations. In *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, pp. 3857–3867, 2026. doi: 10.1109/WACV61042.2026.00376.

Zhaopeng Gu, Bingke Zhu, Zhaowen Li, Guibo Zhu, Yingying Chen, Ming Tang, Peng Su, and Jinqiao Wang. UniVAD v2: Unified visual anomaly detection via support-conditioned boundary construction. *arXiv preprint arXiv:2606.29714*, 2026. URL https://arxiv.org/abs/2606.29714.

Denis Gudovskiy, Shun Ishizaka, and Kazuki Kozuka. CFLOW-AD: Real-time unsupervised anomaly detection with localization via conditional normalizing flows. In *IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022.

Jia Guo, Shuai Lu, Weihang Zhang, Fang Chen, Huiqi Li, and Hongen Liao. Dinomaly: The less is more philosophy in multi-class unsupervised anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. URL https://arxiv.org/abs/2405.14325.

P. Hall. On representatives of subsets. *Journal of the London Mathematical Society*, s1-10(1):26–30, 1935. doi: 10.1112/jlms/s1-10.37.26.

Kengo Ishida, Yuki Takena, Yoshiki Nota, Rinpei Mochizuki, Itaru Matsumura, and Gosuke Ohashi. SA-PatchCore: Anomaly detection in dataset with co-occurrence relationships using self-attention. *IEEE Access*, 11:3232–3240, 2023. doi: 10.1109/ACCESS.2023.3234745.

Soopil Kim, Sion An, Philip Chikontwe, Myeongkyun Kang, Ehsan Adeli, Kilian M. Pohl, and Sang Hyun Park. Few shot part segmentation reveals compositional logic for industrial anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 8591–8599, 2024. doi: 10.1609/aaai.v38i8.28703. URL https://ojs.aaai.org/index.php/AAAI/article/view/28703.

H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1–2):83–97, 1955. doi: 10.1002/nav.3800020109.

Changyi Li, Miao Yu, Kai Dong, and Yu Xiao. Training-free logical and structural anomaly detection via calibrated fusion. arXiv:2609.05091v1, 2026. URL https://arxiv.org/abs/2609.05091v1.

Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. CutPaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

Xurui Li, Ziming Huang, Feng Xue, and Yu Zhou. MuSc: Zero-shot industrial anomaly classification and segmentation with mutual scoring of the unlabeled images. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2401.16753.

Tongkun Liu, Bing Li, Xiao Du, Bingke Jiang, Xiao Jin, Liuyi Jin, and Zhuo Zhao. Component-aware anomaly detection framework for adjustable and logical industrial visual inspection. *Advanced Engineering Informatics*, 58:102161, 2023a. doi: 10.1016/j.aei.2023.102161.

Zhikang Liu, Yiming Zhou, Yuansheng Xu, and Zilei Wang. SimpleNet: A simple network for image anomaly detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023b.

Hiroto Nakata, Yawen Zou, Shunsuke Sakai, Shun Maeda, Chunzhi Gu, Yijin Wei, Shangce Gao, and Chao Zhang. VID-AD: A dataset for image-level logical anomaly detection under vision-induced distraction. *arXiv preprint arXiv:2603.13964*, 2026. URL https://arxiv.org/abs/2603.13964.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, et al. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=a68SUt6zFt. Published version available as arXiv:2304.07193v2.

Yun Peng, Xiao Lin, Nachuan Ma, Jiayuan Du, Chuangwei Liu, Chengju Liu, and Qijun Chen. SAM-LAD: Segment anything model meets zero-shot logic anomaly detection. *Knowledge-Based Systems*, 314:113176, 2025. doi: 10.1016/j.knosys.2025.113176.

Gabriel Peyré and Marco Cuturi. Computational optimal transport: With applications to data science. *Foundations and Trends in Machine Learning*, 11(5-6):355–607, 2019. doi: 10.1561/2200000073.

Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 14318–14328, 2022.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, et al. DINOv3. *arXiv preprint arXiv:2508.10104*, 2025. URL https://arxiv.org/abs/2508.10104v1.

SQUID repository. SQUID: official code repository and DigitAnatomy data, 2026. URL https://github.com/tiangexiang/SQUID/tree/1bb3b70fd6eb66caaf446a68adf8ee6f6caaee63. Pinned repository snapshot; 2026 denotes access year, not dataset publication year.

Michael J. Swain and Dana H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991. doi: 10.1007/BF00130487.

Robin Trombetta and Carole Lartizien. Learning local and global prototypes with optimal transport for unsupervised anomaly detection and localization. In *Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD Workshops)*, pp. 183–201, 2026. doi: 10.1007/978-3-032-19108-3_12.

Chengjie Wang, Wenbing Zhu, Bin-Bin Gao, Zhenye Gan, Jiangning Zhang, Zhihao Gu, Shuguang Qian, Mingang Chen, and Lizhuang Ma. Real-IAD: A real-world multi-view dataset for benchmarking versatile industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Xu Wang, Jingyuan Zhuo, Zhiyuan You, Zhiyu Tan, Yikuan Yu, Siyu Wang, and Xinyi Le. ReinAD: Towards real-world industrial anomaly detection with a comprehensive contrastive dataset. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. Datasets and Benchmarks Track.

Tong Wu, Liang Pan, Junzhe Zhang, Tai Wang, Ziwei Liu, and Dahua Lin. Density-aware Chamfer distance as a comprehensive metric for point cloud completion. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 29088–29100, 2021.

Tiange Xiang, Yixiao Zhang, Yongyi Lu, Alan L. Yuille, Chaoyi Zhang, Tom Weidong Cai, and Zongwei Zhou. SQUID: Deep feature in-painting for unsupervised anomaly detection. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 23890–23901, 2023. doi: 10.1109/CVPR52729.2023.02288.

Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. DRAEM: A discriminatively trained reconstruction embedding for surface anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. DeepEMD: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12200–12210, 2020. doi: 10.1109/CVPR42600.2020.01222.

Jinjin Zhang, Guodong Wang, Yizhou Jin, and Di Huang. Towards training-free anomaly detection with vision and language foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15204–15213, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Zhang_Towards_Training-free_Anomaly_Detection_with_Vision_and_Language_Foundation_Models_CVPR_2025_paper.html. The method introduced there is named LogSAD.

Ximiao Zhang, Min Xu, and Xiuzhuang Zhou. LogiCo: A unified framework for logical and structural anomaly detection. arXiv:2606.28688, 2026. URL https://arxiv.org/abs/2606.28688.

Yang Zou, Jongheon Jeong, Latha Pemula, Dongqing Zhang, and Onkar Dabeer. SPot-the-Difference self-supervised pre-training for anomaly detection and segmentation. In *European Conference on Computer Vision*, 2022.

APPENDIX

A WHERE THE GAIN COMES FROM

The gain comes from count and composition anomalies. A missing or extra part changes the multiset of encoded patches, and it carries the gain. Moving correct parts around should leave that multiset unchanged, and in an exploratory split such rearrangements give no measurable gain. The gain also needs the whole one-to-one rule: neither the ban on reuse nor coverage alone accounts for it (Appendix A.3).

A.1 DEFECT SUBTYPES AND THE PRIMARY CONTRASTS

Every count interval excludes zero and every placement interval covers it (Table 5a). To split the logical stratum we use the defect subtype that LOCO tags on each anomalous image. We call an image a pure rearrangement when its every mask subtype is a placement: swapped compartments (breakfast box), swapped/rotated/misplaced labels and fruit icon (juice bottle), and wrong cable location or flipped connector (splicing connectors). This gives 59 rearrangement images in three categories against 502 count images. The split reuses the per-image scores of Table 5(a) and adds no new data. Used alone as a score on the original references with raw costs, the excess reaches category-macro AUROC 77.45 $[74.89, 80.06]$ logical and 57.00 $[53.41, 60.61]$ structural for DINOv2, and 78.58 $[76.01, 81.14]$ and 58.69 $[55.22, 62.24]$ for DINOv3.

Table 5: **The logical gain comes from count and composition anomalies.** Assignment mean minus NN mean in LOCO category-macro AUROC points. (a) Original references; the last two columns split logical anomalies by subtype, an exploratory split. (b) Encoder-native references, raw costs.

(a) Original references

Configuration	Logical	Structural	Count / composition	Placement only
DINOv2 raw	$+4.96$ $[3.74, 6.30]$	-1.48 $[-2.63, -0.36]$	$+5.29$ $[+4.04, +6.57]$	-0.38 $[-1.68, +0.91]$
DINOv2 pair-median	$+4.96$ $[3.75, 6.30]$	-1.48 $[-2.64, -0.34]$	$+5.23$ $[+3.98, +6.52]$	-0.04 $[-1.32, +1.24]$
DINOv3 raw	$+3.54$ $[2.30, 4.80]$	-2.08 $[-3.12, -1.07]$	$+3.75$ $[+2.52, +5.08]$	-0.38 $[-2.19, +1.45]$
DINOv3 pair-median	$+3.58$ $[2.32, 4.85]$	-2.00 $[-3.05, -0.95]$	$+3.78$ $[+2.52, +5.12]$	-0.48 $[-2.52, +1.62]$

(b) Encoder-native references, raw costs, corrected intervals

DINOv2 raw	$+4.68$ $[2.94, 6.47]$	-1.82 $[-3.53, -0.20]$
DINOv3 raw	$+3.57$ $[1.99, 5.29]$	-1.66 $[-3.07, -0.27]$

A.2 EVERY RULE, THE ENTROPIC RELAXATION, AND EXCESS ALONE

Lightly softened, the assignment keeps a positive logical contrast; heavily softened, the contrast reverses (Table 6b). Part (a) lists each rule under pair-median costs.

Table 6: **A small entropic relaxation keeps a positive logical contrast; a large one reverses it.** (a) Every score rule under pair-median costs, LOCO category-macro AUROC (%), original references, as Table 2 does for raw costs; only the first two rows differ in the matching rule alone. Log./Str.: logical/structural anomalies. (b) Sinkhorn minus reusable NN in logical category-macro AUROC points on the references of Table 5(a), with family-corrected 95% intervals.

(a) Every score rule, pair-median costs

	DINOv2		DINOv3	
Score	Log.	Str.	Log.	Str.
NN mean (reuse allowed)	73.16	69.54	77.07	72.76
Assignment mean (one-to-one)	78.12	68.06	80.65	70.76
Directed maximum	63.40	69.79	68.43	77.70
Min-sum assignment: max	71.75	60.88	77.53	66.62
Partial assignment (95%)	76.87	66.04	77.56	65.79
Chamfer mean	74.73	67.84	79.11	71.71
Hausdorff maximum	69.12	68.36	77.93	77.81

(b) Entropic relaxation (Sinkhorn)

Encoder	ε	Logical difference
DINOv2	0.1	$+1.91$ $[0.41, 3.41]$
DINOv2	1.0	-12.92 $[-16.72, -8.96]$
DINOv3	0.1	$+1.79$ $[0.42, 3.17]$
DINOv3	1.0	-18.19 $[-22.29, -13.90]$

A.3 WHICH HALF OF THE ONE-TO-ONE RULE DOES THE WORK

A one-to-one assignment puts two constraints on the reference: no patch may be used twice, and every patch must be used. On LOCO, the second constraint alone recovers half or more of the logical gain. On VisA that coverage arm changes sign while the full rule keeps its DINOv2 gain, so the LOCO split between the two constraints does not carry over. We separate them with the four cells below, re-reducing the per-image costs recorded for the main results (same strata, category macro and 20,000-draw bootstrap). We ran no new inference.

The four cells. We write C_{ij} for the cost of matching query patch i to reference patch j. The reusable nearest-neighbor mean $\mathrm{NN} = \mathrm{mean}_i \min_j C_{ij}$ allows reuse and does not require coverage. The assignment mean forbids reuse and requires coverage. The two remaining cells are a capacity constraint, which caps reuse without requiring coverage, and

$$s_{\mathrm{cover}} \;=\; \mathrm{mean}_j \min_i C_{ij} \;=\; 2\,(\text{Chamfer mean}) - \mathrm{NN}, \tag{5}$$

which requires coverage while still allowing reuse. The minimum-cost reuse-allowed covering decouples over reference patches, so (5) is exact and needs no solver.

Table 7: **Coverage alone recovers about half the logical gain on LOCO and changes sign on VisA.** Category-macro AUROC change against the reusable nearest-neighbor mean, with 95% bootstrap intervals, for the full one-to-one rule and for coverage with reuse allowed; five LOCO categories and eight VisA quantity categories, encoder-native references as in Table 5(b). Table 18 uses the seven VisA categories eligible under a stricter rule, so its values differ. Coverage alone pays more on structural anomalies.

Benchmark	Encoder	One-to-one	Coverage only
LOCO logical	DINOv2	$+4.68$ $[+3.33, +6.14]$	$+2.43$ $[+0.90, +4.06]$
	DINOv3	$+3.57$ $[+2.34, +4.88]$	$+2.18$ $[+0.98, +3.46]$
LOCO structural	DINOv2	-1.82 $[-3.14, -0.53]$	-4.10 $[-5.85, -2.34]$
	DINOv3	-1.66 $[-2.74, -0.59]$	-2.86 $[-3.99, -1.75]$
VisA quantity	DINOv2	$+1.64$ $[+0.67, +2.68]$	-0.62 $[-2.14, +0.80]$
	DINOv3	-0.74 $[-1.38, -0.13]$	-0.99 $[-1.71, -0.31]$

The coverage–reuse decomposition. When we force coverage while still allowing reuse (Table 7), we recover 0.52 of the DINOv2 logical gain and 0.61 of the DINOv3 gain, while the structural cost grows to about 2.2 and 1.7 times that of the full assignment. Neither half of the constraint carries the gain alone: coverage without the ban on reuse buys part of it at a larger structural price, and a reuse cap of two, which drops the coverage requirement, keeps almost none of it.

The coverage arm across benchmarks. The one-to-one rule, which constrains both sides, carries over from LOCO to the VisA quantity categories for DINOv2. The coverage arm does not: it is positive on LOCO and negative on VisA. NN scores query patches and s_{cover} scores reference patches, so the coverage arm responds to content present in the reference and absent from the query, and would be expected to help on missing-part anomalies and not on added-part ones. A missing/added asymmetry is one candidate explanation for the sign change: 181 of LOCO's 561 logical-anomaly images carry a missing-part subtype (Appendix B.2).

Table 8: **A second use per reference patch removes almost all of the gain.** LOCO logical category-macro AUROC change over the five categories against the NN mean when each reference patch may be used at most k times, encoder-native references; $k = 1$ is the one-to-one assignment and large k is the NN mean. Entries of 0 are exact.

Encoder	k=1	k=2	k=3	k=4	k=6	k=8	k=12	k=16	k=24	k=32
DINOv2	$+4.68$	$+0.29$	$+0.05$	-0.00	-0.02	-0.00	0	0	0	0
DINOv3	$+3.57$	$+0.20$	$+0.03$	$+0.01$	0	0	0	0	0	0

The reuse-cap sweep. When we raise the cap from one to two (Table 8), the arm retains only 6% of the DINOv2 gain and 6% of the DINOv3 gain, and by $k \geq 12$ the arm is numerically identical to the baseline. The two identities that bound the sweep hold exactly: $k = 1$ reproduces the assignment

mean and large k reproduces the nearest-neighbor mean. Almost all of the gain sits at $k = 1$, the only capacity at which every reference patch must be matched. This is the same coverage constraint isolated in Table 7, approached from the capacity side.

Excess by encoder. DINOv3 pays less excess than DINOv2 (Table 9).

Table 9: **DINOv3 pays less excess than DINOv2 on both benchmarks.** Excess relative to the nearest-neighbor mean on normal images, and the AUROC (%) of the excess used alone as a score, for every stratum, encoder-native references (Section 4.2 gives original-reference values). On VisA count anomalies the DINOv3 excess is also less discriminative.

		Excess / NN		Excess alone	
Benchmark	Stratum	DINOv2	DINOv3	DINOv2	DINOv3
LOCO	logical	0.2616	0.1993	77.69	78.82
	structural	0.2616	0.1993	56.41	61.14
VisA	quantity	0.2666	0.1838	87.24	84.42
	structural	0.2625	0.1880	61.07	62.79
	mixed	0.2678	0.1781	80.85	77.40

Excess by encoder on VisA. On normal images the ratio of excess to nearest-neighbor cost lies in $[0.2616, 0.2678]$ for DINOv2 and in $[0.1781, 0.1993]$ for DINOv3, with no overlap between the encoders on either benchmark. The DINOv3 excess still carries logical signal: on LOCO it reaches 78.82 on its own, against its encoder-native nearest-neighbor baseline of 77.92. On VisA it is smaller and is added to a stronger baseline (93.27 against 91.55), which leaves it less room to help.

B When the gain holds

Scored against one complete reference at a time, the gain holds. Pooling references weakens it, and so do incomplete or contaminated ones. When both arms reduce over the same bank, a structural cost appears.

B.1 Pooled, per-reference and matched-reducer banks

Pool five references and the contrast nearly vanishes in both strata. Score them one at a time and it returns (Table 10a). Pooled, all four intervals include zero. The paired contrast falls from $K = 1$ to $K = 5$ by 4.44 points for DINOv2 ($[-5.83, -3.08]$) and 3.65 for DINOv3 ($[-4.98, -2.39]$). At $K = 5$ the pooled assignment-minus-Chamfer contrasts stay at -1.54 and -1.56. When both arms reduce over the same complete R-reference bank, the logical gain holds at every bank size and a structural cost appears under DINOv3 (Table 10c). There 5 of 12 structural intervals fall wholly below zero (†), all under DINOv3. Every DINOv2 interval crosses zero.

B.2 Incomplete and contaminated references

A reference that is itself missing a part nearly erases the logical gain, and anomalous images in the bank wear it down steadily without removing it (Table 11). These are the two failure modes of Section 3: the count bound is symmetric, so under the same premises an incomplete reference makes normal images pay excess too.

Setup. We use DINOv2, raw costs, encoder-native retrieval, all five LOCO categories and all 1,568 test images, scored as in Appendix F.3 from the stored patch-token features. For the incomplete references we take 181 real LOCO logical-anomaly images whose ground-truth mask marks a missing-part defect subtype (41 breakfast box, 41 juice bottle, 49 pushpins, 24 screw bag, 26 splicing connectors) and retrieve them in place of the normal bank at a fixed $N = 256$ tokens in both query and reference. To contaminate the bank, we inject a fraction ρ of that category's anomalous test images into it. We never inject the 575 normal test images, and we always exclude a query from its own bank.

Table 10: **Pooling removes the contrast; scoring one reference at a time restores it.** Assignment minus reusable NN in LOCO category-macro AUROC points over five categories. (a) One reference, five pooled references, and the minimum over five complete references, where each method may pick a different reference. Pooled-five intervals are adjusted for a family of four. (b) Per-reference banks against pooled NN, behind Table 3. (c) Both arms reduced over the same bank of R complete references; † marks a structural interval wholly below zero. Under matched reduction the structural cost appears for DINOv3.

(a) One reference, pooled five, and complete five

Encoder	One reference	Pooled five: logical	Pooled five: structural	Complete five: min
DINOv2	$+4.68$	$+0.24$ $[-0.04, +0.55]$	$+0.03$ $[-0.27, +0.33]$	$+5.69$ $[+4.20, +7.22]$
DINOv3	$+3.57$	-0.08 $[-0.27, +0.11]$	-0.01 $[-0.23, +0.20]$	$+3.66$ $[+2.45, +4.91]$

(b) Per-reference banks against the pooled bank

Encoder	R	Logical	Structural
DINOv2	1	$+6.13$ $[4.78, 7.63]$	-0.37 $[-1.46, 0.69]$
	2	$+7.25$ $[5.53, 8.99]$	-0.79 $[-2.22, 0.67]$
	4	$+7.22$ $[5.39, 9.11]$	-0.67 $[-2.24, 0.87]$
	8	$+5.77$ $[3.89, 7.59]$	-2.39 $[-4.21, -0.59]$
DINOv3	1	$+5.58$ $[4.11, 7.18]$	-1.68 $[-2.87, -0.49]$
	2	$+5.63$ $[3.87, 7.40]$	-2.68 $[-4.37, -1.10]$
	4	$+5.02$ $[3.31, 6.80]$	-4.94 $[-6.74, -3.33]$
	8	$+0.55$ $[-1.17, 2.14]$	-6.84 $[-8.88, -4.80]$

(c) Matched reducers

Encoder	Reducer	R	Logical ↑	Structural	Holm p
DINOv2	min	2	$+7.15$ $[5.57, 8.76]$	$+0.01$ $[-1.24, +1.38]$	1.000
	mean	2	$+6.97$ $[5.40, 8.57]$	-0.32 $[-1.70, +1.13]$	1.000
	min	4	$+6.28$ $[4.84, 7.83]$	-0.41 $[-1.85, +1.08]$	1.000
	mean	4	$+5.99$ $[4.29, 7.61]$	$+0.69$ $[-0.66, +2.08]$	0.945
	min	8	$+6.71$ $[4.99, 8.38]$	-1.05 $[-2.34, +0.26]$	0.387
	mean	8	$+5.71$ $[4.18, 7.35]$	-0.15 $[-1.75, +1.62]$	1.000
DINOv3	min	2	$+7.10$ $[5.46, 8.88]$	-1.07 $[-2.29, +0.15]$	0.084
	mean	2	$+5.31$ $[3.85, 6.80]$	-2.19 $[-3.50, -0.83]^{\dagger}$	0.006
	min	4	$+5.54$ $[4.00, 7.19]$	-2.56 $[-3.87, -1.27]^{\dagger}$	0.000
	mean	4	$+5.80$ $[4.31, 7.36]$	-1.74 $[-2.97, -0.50]^{\dagger}$	0.007
	min	8	$+4.42$ $[3.03, 5.88]$	-2.50 $[-3.75, -1.28]^{\dagger}$	0.000
	mean	8	$+4.54$ $[3.08, 5.97]$	-2.08 $[-3.53, -0.65]^{\dagger}$	0.006

Table 11: **An incomplete reference nearly removes the logical gain, and contamination lowers it gradually.** Assignment minus reusable NN in LOCO category-macro AUROC points, DINOv2 raw costs, encoder-native references, 95% intervals; ρ is the injected fraction of anomalous test images. Control row: Table 5(b).

Reference bank	Logical	95%	Structural	95%
Complete normal reference (control)	$+4.68$	$[+3.33, +6.14]$	-1.82	$[-3.14, -0.53]$
Reference missing a part	$+0.96$	$[-0.83, +2.75]$	-2.61	$[-4.50, -0.72]$
Bank contaminated, $\rho = 0.05$	$+4.45$	$[+3.06, +5.93]$	-1.54	$[-2.84, -0.23]$
$\rho = 0.10$	$+3.77$	$[+2.39, +5.21]$	-1.81	$[-3.24, -0.39]$
$\rho = 0.25$	$+3.61$	$[+2.12, +5.11]$	-2.05	$[-3.46, -0.63]$
$\rho = 0.50$	$+2.55$	$[+0.99, +4.13]$	-0.96	$[-2.43, +0.48]$

Table 12: **Without breakfast box the logical gain still clears zero in seven of eight cells.** Assignment mean minus reusable NN mean in LOCO category-macro AUROC points. *Macro* is the main-text row; *median* is the median of the five per-category deltas; *omit largest* drops the category with the largest positive logical delta (breakfast box in every cell), with 95% intervals; *pos.* counts positive per-category deltas. The median category gain is positive in all eight.

		Logical anomalies				Structural anomalies			
Protocol	Configuration	Macro	Median	Omit largest	pos.	Macro	Median	Omit largest	pos.
Original	DINOv2 raw	$+4.96$	$+2.96$	$+3.01$ $[+1.75, +4.26]$	5/5	-1.48	-1.26	-1.55 $[-2.80, -0.33]$	1/5
Original	DINOv2 pair-median	$+4.96$	$+3.02$	$+3.08$ $[+1.81, +4.32]$	5/5	-1.48	-1.38	-1.53 $[-2.79, -0.30]$	1/5
Original	DINOv3 raw	$+3.54$	$+1.27$	$+0.98$ $[+0.10, +1.89]$	5/5	-2.08	-1.87	-2.23 $[-3.29, -1.13]$	1/5
Original	DINOv3 pair-median	$+3.58$	$+1.28$	$+0.89$ $[-0.01, +1.83]$	4/5	-2.00	-1.84	-2.05 $[-3.20, -0.88]$	1/5
Encoder-native	DINOv2 raw	$+4.68$	$+3.66$	$+3.07$ $[+1.69, +4.49]$	5/5	-1.82	-1.82	-1.83 $[-3.18, -0.51]$	1/5
Encoder-native	DINOv2 pair-median	$+4.55$	$+3.42$	$+2.97$ $[+1.56, +4.42]$	5/5	-2.00	-2.32	-1.93 $[-3.32, -0.54]$	1/5
Encoder-native	DINOv3 raw	$+3.57$	$+1.35$	$+1.18$ $[+0.24, +2.11]$	5/5	-1.66	-1.33	-1.92 $[-3.04, -0.81]$	1/5
Encoder-native	DINOv3 pair-median	$+3.64$	$+1.23$	$+1.13$ $[+0.17, +2.10]$	4/5	-1.62	-1.59	-1.84 $[-3.00, -0.69]$	1/5

B.3 Concentration in one category

Breakfast box carries the largest logical gain in every configuration, $+10.88$ to $+14.32$ across the eight cells of Table 12, against -0.04 to $+6.34$ for the other four categories. Without it the gain shrinks, most for DINOv3, but stays positive. The median category gain is positive in all eight cells, and without breakfast box seven of the eight logical intervals still exclude zero. In the eighth, the original-reference DINOv3 pair-median cell, the interval reaches zero while its encoder-native counterpart excludes it. The structural loss is just as broad: four of five categories are negative in every cell, and the omission leaves it between -1.53 and -2.23.

For each encoder and cost normalization we re-reduce the frozen per-image records of Table 5. We report the median of the five per-category deltas, the macro after omitting the category with the largest positive logical delta, and how many deltas are positive. We chose the omission rule on the logical stratum and applied it unchanged to the structural one. It removes breakfast box in all eight cells.

C Excess in trained detectors

Added to a trained detector, conditional excess raises logical AUROC on both hosts. On EfficientAD-M it does so in every training seed and stays inside the structural harm margin. On PatchCore it pays more structural AUROC than the margin allows.

C.1 PatchCore

Conditional excess lifts PatchCore from 79.26 to 84.54 logical AUROC and beats the host, directed-mean and Chamfer arms, every bound above zero. It costs 1.4 structural points, beyond the -1 harm margin (Table 13). On CAD-SD the host starts low, so the capacity-specific part of the count gain is its margin over the two coverage arms.

For PatchCore we use recovered Anomalib 2.2.0 (Akcay et al., 2022) WideResNet50 checkpoints (layers 2, 3) with stored preprocessing and the raw nearest-neighbor score. CAD-SD's official release lists 400 normal training images; 80 are exact duplicates of others, leaving 320 distinct. We draw the normal bank of 192 of these under the official 10% greedy coreset and withhold 128. The added branch uses DINOv2 final normalized patch tokens at 224×224, native mean-descriptor retrieval, and exact mean assignment on raw squared-Euclidean costs: excess is assignment mean minus directed mean, and expected excess is the median over its 32 nearest fitting normals in directed-mean cost. We split LOCO validation normals and withheld CAD normals two-thirds/one-third for fitting and scaling. The fused score adds the positive standardized residual to the standardized baseline at unit weight, and the three simpler additions use the same calibration. No anomaly label sets the formula or the weight, and calibration, fitting, bank, and test splits are disjoint within each category.

Table 13: **On PatchCore the excess arms gain logical and count AUROC, but the LOCO structural cost exceeds the harm margin.** Left: image AUROC (%) of five integration arms; LOCO is an equal-category macro. Right: conditional excess minus comparator in AUROC points; a criterion is met when its lower bound exceeds its margin.

	LOCO		CAD-SD	
Score	Logical	Structural	Count	Structural
PatchCore	79.26	88.41	56.48	98.29
+ directed mean	81.36	86.98	95.38	97.52
+ Chamfer mean	81.85	87.12	96.74	97.58
+ excess	84.51	87.14	99.70	98.00
+ conditional excess	84.54	87.03	100.00	97.39

Population	Comparator	Δ	Lower	Margin	Criterion
LOCO logical	PatchCore	+5.28	+2.96	+0	**Met**
LOCO logical	directed mean	+3.18	+0.89	+0	**Met**
LOCO logical	Chamfer mean	+2.69	+0.62	+0	**Met**
LOCO logical	excess	+0.03	−0.13	+0	Not established
LOCO structural	PatchCore	−1.38	−2.80	−1	Not established
CAD-SD count	PatchCore	+43.52	+34.89	+0	**Met**
CAD-SD count	directed mean	+4.62	+1.71	+0	**Met**
CAD-SD count	Chamfer mean	+3.26	+1.20	+0	**Met**
CAD-SD count	excess	+0.30	+0.00	+0	Not established
CAD-SD structural	PatchCore	−0.89	−2.63	−1	Not established

C.2 EFFICIENTAD-M

Conditional excess raises EfficientAD-M's logical AUROC and stays within the structural margin. It clears every bound but that against simple excess, which scores nearly the same. Here the conditioning adds little (Table 14).

Our host is Anomalib 2.6.0 (Akcay et al., 2022) EfficientAD medium, which we train for 70 epochs per LOCO category. This host reaches 76.56/87.69 logical/structural LOCO AUROC. Batzner et al. (2024) report 86.8/94.7 (their Table 2) for the same architecture trained for the official ~70,000 steps; our host follows the library default of 70 epochs per category instead. Every arm in Table 4 is fused onto this one trained host and scored on the same split, so the conditional-excess gain is a within-host effect that does not depend on how strongly the host itself was trained. The native scalar is the spatial maximum of the averaged student–teacher and student–autoencoder maps before postprocessing. Fitting, scaling, and test splits are disjoint: 203 fitting and 102 scaling normals from 305 validation normals, with per-category fit/scale/test counts 41/21/275, 36/18/330, 46/23/310, 40/20/341, and 40/20/312. Every arm shares these normals and this recipe. Writing h for the host score, e for the assignment excess (Equation 7), and $\mathcal{N}_{32}(x)$ for the 32 fitting normals closest to x in reusable mean s_{reuse}, the conditional statistic and fused score are

$$e_{\text{cond}}(x) = e(x) - \underset{n \in \mathcal{N}_{32}(x)}{\text{median}}\, e(n), \qquad s_{\text{fused}}(x) = \frac{h(x) - \mu_h}{\sigma_h} + \max\Big(0,\ \frac{e_{\text{cond}}(x) - \mu_c}{\sigma_c}\Big), \quad (6)$$

Each μ is a median and each σ the raw MAD over the scaling normals. The other arms substitute their own statistic under the same calibration, and substituting e recovers the simple-excess arm. Seeds 0 to 4 give five EfficientAD-M models, the first being the run reported above. Taking the training run as the independent unit with $t_{0.975,4}$ intervals, the logical lower bound stays positive and the structural lower bound stays above -1 point (Table 4).

Table 14: **On EfficientAD-M, conditional excess beats every arm but simple excess, which scores nearly the same.** Conditional excess minus the named comparator in AUROC points, with 95% intervals and adjusted lower bounds; a logical row must clear 0, the structural row -1.

Stratum	Comparator	Difference	95% interval	Adj. lower	Bound
Logical	Baseline	+7.70	[+5.78, +9.74]	+5.42	Clears
Logical	+ Reusable	+3.75	[+2.02, +5.53]	+1.68	Clears
Logical	+ Chamfer	+3.22	[+1.65, +4.84]	+1.38	Clears
Logical	+ Simple excess	+0.07	[−0.04, +0.18]	−0.06	Does not clear
Structural	Baseline	+0.71	[−0.37, +1.85]	−0.55	Clears

Omitting each category. We omit each LOCO category in turn from the five-seed result, with the training run as the independent unit (Table 15).

Every logical interval stays above zero. The weakest lower bound, +4.49, comes without breakfast box. The same omission turns the structural change to −0.65 [−0.85, −0.46]: on the other four categories the fused detector pays a small structural cost, inside the −1-point harm margin.

Table 15: **Dropping any one category leaves the logical gain above zero.** Leave-one-category-out on EfficientAD-M: conditional excess minus baseline in category-macro AUROC over the five LOCO categories, with 95% t-intervals over five training seeds, encoder-native references. Without breakfast box the structural change becomes a small cost inside the -1-point harm margin.

Omitted category	Logical	Structural
None (main text)	$+7.44$ $[+6.68, +8.19]$	$+0.53$ $[+0.06, +0.99]$
Breakfast box	$+5.74$ $[+4.49, +6.99]$	-0.65 $[-0.85, -0.46]$
Juice bottle	$+9.30$ $[+8.37, +10.22]$	$+0.65$ $[+0.07, +1.23]$
Pushpins	$+7.25$ $[+6.87, +7.64]$	$+1.35$ $[+0.87, +1.83]$
Screw bag	$+7.63$ $[+6.70, +8.57]$	$+0.51$ $[-0.02, +1.05]$
Splicing connectors	$+7.25$ $[+6.56, +7.95]$	$+0.77$ $[+0.21, +1.33]$

Neighborhood size. When we change only k, from 4 up to the fitting-set cap of 36, and recompute the fusion from the frozen per-image scores, the logical gain moves by 0.24 AUROC points and the structural change stays positive (Table 16).

Table 16: **The detector gain barely moves with neighborhood size.** Fused conditional excess minus baseline for seed-0 EfficientAD-M in LOCO category-macro AUROC points, recomputed from frozen scores, varying only the number k of nearest fitting normals; shading marks the value used throughout.

k	4	8	16	24	32	36
Logical	$+7.51$	$+7.65$	$+7.46$	$+7.53$	$+7.70$	$+7.68$
Structural	$+0.42$	$+0.83$	$+0.90$	$+0.79$	$+0.71$	$+0.73$

C.3 Retraining the seed-0 detector

Retraining the seed-0 EfficientAD-M gives 83.56 logical macro AUROC for the fused detector, against 84.26 in the original seed-0 run, and a gain over Chamfer of $+2.89$ against $+3.22$. The gain keeps its sign and moves by less than a point, more than the rounding our reproduction rule allowed, so the retrained model misses the original values. The fused score uses the original matching costs, so only the detector differs.

C.4 Low-FPR endpoint

All four LOCO logical excess-minus-Chamfer contrasts favor excess, with lower bounds that reach zero. Our endpoint (Table 17) is the average TPR over FPR 0 to 0.05, that is, 100 times the ROC area there divided by 0.05. PatchCore's simple and conditional contrasts are $+3.80$ $[+0.08, +7.64]$ and $+3.83$ $[+0.07, +7.60]$; EfficientAD-M's are $+2.65$ $[-0.12, +5.39]$ and $+2.65$ $[-0.21, +5.35]$. Their family-corrected lower bounds, -0.71, -0.76, -0.72 and -0.84, all reach zero.

Table 17: **At low false-positive rates the excess arms lead on logical and count anomalies and trail Chamfer on LOCO structural ones.** Average TPR (%) over FPR 0–5% for every detector arm; LOCO is an equal-category macro, and CAD-SD is a secondary single-family analysis.

Detector	Population	Stratum	Base	+Reuse	+Chamfer	+Excess	+Cond.
PatchCore	LOCO	Logical	40.67	47.54	47.94	51.74	51.77
PatchCore	LOCO	Structural	54.44	53.47	53.83	50.51	49.80
EfficientAD-M	LOCO	Logical	39.34	43.62	45.05	47.70	47.69
EfficientAD-M	LOCO	Structural	52.54	53.26	54.59	53.59	53.52
PatchCore	CAD-SD	Count	2.92	50.77	57.34	94.05	100.00
PatchCore	CAD-SD	Structural	81.97	77.18	77.76	80.75	78.14

D Count anomalies beyond LOCO

Outside LOCO the count-anomaly gain appears on CAD-SD, DigitAnatomy and VID-AD, and on VisA for DINOv2 (Table 18). Under DINOv3 it reverses on VisA, where the excess is less discrimi-

native and the baseline stronger (Appendix A.3). On Real-IAD the structural sign follows the logical sign in all but two cells.

D.1 CAD-SD, DigitAnatomy, VID-AD and VisA

CAD-SD and DigitAnatomy. For both we reuse existing DINOv2 ViT-L/14 and WRN50 layer-3 features at 224 pixels (256 and 196 tokens, dimension 1,024). We keep CAD-SD's exact-deduplication correction: 320 normal training references, and 168 normal, 40 lacking, 44 over-coupling, 41 paint, and 41 scratch test images, with no train/test byte overlap (count pools lacking and over-coupling; structural pools paint and scratch). DigitAnatomy has 500 normal references, 500 normal tests, and 100 author-generated missing-component tests. Retrieval and costs follow Appendix F.3.

VID-AD. We keep all 50 VID-AD tasks, ten scenarios and five conditions (7,895 test queries): a 45-image normal bank per task, five calibration normals excluded, 50 negatives per task. Quantity-only anomalies are primary in Cookies, Fruits, Sticks, and Tools; conditions average within scenario, and mixed Quantity+X anomalies lie outside this endpoint.

VisA. We stratify count anomalies from structural ones on the seven eligible categories and report all twelve for the all-anomaly column.

Table 18: **The count gain holds on three more datasets; on VisA it holds for DINOv2 and reverses for DINOv3.** One-to-one mean minus NN mean in AUROC points. (a) Raw costs; CAD-SD and DigitAnatomy (synthetic missing components) use WideResNet-50 (WRN50) and DINOv2; CAD-SD structural effects are unresolved, and VID-AD quantity anomalies at one reference carry lower bounds. (b) VisA macro image AUROC under raw and pair-median costs.

(a) CAD-SD, DigitAnatomy and VID-AD

Population	Encoder or contrast	Difference	Interval or lower bound
CAD-SD count	DINOv2	$+0.28$	$[+0.02, +0.83]$
CAD-SD count	WRN50	$+2.70$	$[+1.20, +4.73]$
CAD-SD structural	DINOv2	-0.81	$[-2.71, +1.00]$
CAD-SD structural	WRN50	$+0.70$	$[-1.21, +2.85]$
DigitAnatomy missing	DINOv2	$+6.46$	$[+3.84, +9.33]$
DigitAnatomy missing	WRN50	$+22.60$	$[+19.57, +25.74]$
VID-AD quantity	assignment–reuse	$+10.64$	$\geq +7.14$
VID-AD quantity	assignment–Chamfer	$+9.98$	$\geq +7.04$

(b) VisA

Configuration	Quantity	Structural	All 12
DINOv2 raw	$+1.87$ $[+0.74, +3.05]$	-0.79 $[-1.79, +0.19]$	-0.74 $[-1.53, +0.02]$
DINOv3 raw	-0.84 $[-1.54, -0.14]$	-2.79 $[-3.78, -1.88]$	-2.29 $[-2.96, -1.64]$
DINOv2 pair-median	$+1.95$ $[+0.73, +3.22]$	-0.64 $[-1.66, +0.35]$	-0.66 $[-1.39, +0.09]$
DINOv3 pair-median	-0.83 $[-1.61, -0.09]$	-2.57 $[-3.53, -1.64]$	-1.92 $[-2.55, -1.27]$

D.2 Real-IAD

In 16 of 18 cells structural AUROC moves the same way as logical AUROC, with an interval excluding zero. Our sign criterion needs all 18, so it is not met. The exceptions are min-over-references at $R = 5$ (DINOv2 $+0.131$ $[-0.079, +0.346]$, DINOv3 $+0.091$ $[-0.105, +0.303]$).

We run three arms against the per-image one-to-one baseline: one-to-one NN, min-over-references, and mean-over-references, at $R \in \{2, 5, 8\}$ for both encoders. The logical endpoint is the original run. We re-derive the structural endpoint from the same stored scores, over the 24 categories with a non-empty structural stratum (*mint* has none), with a 2,000-draw category bootstrap at each R. Representative cells: DINOv2 mean-over-references at $R = 8$ gives -7.091 $[-7.426, -6.754]$ structural against -7.294 logical, and DINOv3 one-to-one NN at $R = 2$ gives -0.191 $[-0.249, -0.132]$ structural against -0.068 logical. These populations reuse previously seen images, so the check tests only the sign of the tradeoff, not the LOCO effect sizes.

E PROOFS

E.1 SEPARATION BOUNDS

Under the main-text type assumptions, at most $\min(n_t, m_t)$ occurrences of type t can be matched within type, so at least $N - \sum_t \min(n_t, m_t) = \frac{1}{2}\sum_t |n_t - m_t|$ edges cross types, each costing at least δ; dividing by N proves the bound. In the terms of histogram intersection (Swain & Ballard, 1991), the bound is δ times one minus the normalized intersection of the two type counts. Identical support supplies a zero-cost partner in each direction, proving the coverage statement. Whether learned contextual features keep those types is not tested. Section 4.3 measures whether a pooled bank keeps single-image counts. The bound does not cover arbitrary independent-token scorers, whose nonzero-distance mean can change under repetition, and says nothing about semantic counting or localization.

Propositions 2 and 3 use squared-Euclidean costs and equal cardinality N. For substitution, the query is $\mathbf{R}$ with r_a replaced by r'_b. Each unchanged query token r_j has itself in $\mathbf{R}$ at zero cost, and r'_b has r_b at cost at most δ^2, so every directed (query-to-reference) reusable per-token cost is at most δ^2; two-way scores such as Chamfer also pay on the column of r_a and are not covered. A mass-preserving rule must send mass $1/N$ into the column of r_a. Every unchanged token satisfies $\|r_j - r_a\| \geq \gamma_a$ for $j \neq a$, and the triangle inequality gives $\|r'_b - r_a\| \geq \|r_b - r_a\| - \|r'_b - r_b\| \geq \gamma_a - \delta$. Every entry of that column therefore costs at least $\max(\gamma_a - \delta, 0)^2$, and since all other costs are nonnegative the total cost is at least $\max(\gamma_a - \delta, 0)^2/N$. The argument uses only the column marginal, so it covers the optimal assignment and any feasible transport plan. For a retrieved bank scored one reference at a time, $\gamma(\mathbf{Q}, \mathbf{R}) \geq m$ means that some token r_j of that reference is at distance at least m from every query patch $\mathbf{q}_i$, and the same column argument gives a mean cost of at least m^2/N. The argument needs every column to receive mass, so it does not apply to a pooled bank, whose spare columns are unpenalized. Shape contexts use dummy nodes so outliers go unmatched (Belongie et al., 2002). A pooled bank gives a surplus part real normal patches to match, at their own cost.

E.2 RESIDUAL COSTS AND THE RANKING IDENTITY

Writing $m_i = \min_j C_{ij}$ for the row minima,

$$e(C) = s_{\text{bij}}(C) - s_{\text{reuse}}(C), \qquad Ne(C) = \min_{\pi \in \mathcal{S}_N} \sum_i (C_{i,\pi(i)} - m_i), \tag{7}$$

which follows by subtracting the fixed row minima from every assignment. Every residual is nonnegative, so $e(C) \geq 0$, with equality exactly when the graph of row-minimizing edges has a perfect matching; in Hall's formulation, every subset of query rows must collectively own at least as many distinct minimizing columns (Hall, 1935), counting all tied minima. The excess is unchanged by adding a constant to each row. Every observed pair in our study has positive excess, so discrimination depends on its magnitude rather than on the zero criterion. For anomaly a and normal image n,

$$s_{\text{bij}}(a) - s_{\text{bij}}(n) = [s_{\text{reuse}}(a) - s_{\text{reuse}}(n)] + [e(a) - e(n)], \tag{8}$$

so the constraint repairs a normal–anomaly pair when the excess difference exceeds a negative nearest-neighbor margin and breaks it when it exceeds a positive one; a larger excess alone is not enough. Counting repaired and broken pairs with ties at half credit recovers the AUROC change. Within each category the identity is exact: the AUROC change is the net fraction of normal–anomaly pairs whose order the constraint corrects, and the reported change is the equal-weight mean of these fractions. It does not close pooled across categories, because the reported figure is a category macro average and the categories contribute different numbers of pairs.

F SCORES, FEATURES AND REFERENCES

We score every rule on the same realized cost matrices, built from frozen features and a fixed set of references.

System	Matching unit	Reuse allowed	Segmenter or detector	Learned or trained	Reference	Rule varied alone
LogSAD (Zhang et al., 2025)	patch; SAM region	patch: yes; region: no (Hungarian)	SAM, CLIP	nothing	pooled bank	no
SAM-LAD (Peng et al., 2025)	SAM object	no (partial, Sinkhorn with dustbin)	SAM	pretrained dustbin and DCGA weights; none per scene	k retrieved images, each matched	no
ObjectCore (Fučka et al., 2026)	object; patch	object: no (padded Hungarian); patch: yes	Grounding DINO, SAM	detector fine-tuned on the support set	one retrieved image; pooled patch bank	no
HETMM (Chen et al., 2025)	pixel within patch window	yes, both directions	none	nothing (templates selected)	pooled template set	direction only; reuse kept
OT prototypes (Trombetta & Lartizien, 2026)	patch to prototype	yes at test; OT in training only	none	prototypes	prototypes from the training set	no
UniVAD v2 (Gu et al., 2026)	patch token; component	yes (NN; one-sided softmax)	component partition	transport module, head, gate	support set	no
PatchCore (Roth et al., 2022)	patch	yes	none	nothing (coreset selected)	pooled coreset bank	n/a (NN only)
Anomaly-DINO (Damm et al., 2025)	patch	yes	optional foreground mask	nothing	pooled k-shot bank	n/a (NN only)
Ours	patch	no (exact assignment)	none	nothing	one complete image	yes

Table 19: **Prior logical-anomaly matchers change the matching rule together with other components.** LogSAD, SAM-LAD and ObjectCore forbid reuse only between segmented regions or detected objects, UniVAD v2 and the OT prototypes add trained parts, and HETMM changes the direction of matching but keeps reuse. The last column asks whether an experiment in the cited paper swaps the rule while features, reference and scoring stay fixed. Here only the matching rule changes, on one patch-cost matrix and a single complete reference; pooled and per-reference banks are controls (Section 4.3).

F.1 What prior matchers change with the rule

PatchCore and AnomalyDINO keep nearest-neighbor matching throughout; every other prior system below changes more than the matching rule (Table 19).

F.2 Score definitions and transport

Our max-edge control solves a minimum-sum assignment and reports its largest selected cost,

$$\pi^* \in \arg\min_{\pi \in \mathcal{S}_N} \sum_i C_{i,\pi(i)}, \qquad s_{\mathrm{A}} = \max_i C_{i,\pi^*(i)}, \tag{9}$$

and we solve it by exact linear assignment (Kuhn, 1955), not minimum-bottleneck assignment. Our partial rule minimizes total cost over $k = \mathrm{round}(0.95N)$ disjoint pairs and reports their maximum. It discards unmatched occurrences, so it relaxes coverage as well as capacity. Reordering rows or columns leaves the assignment total unchanged, and when we swap tokens in the intervention arrays every rule gives AUROC 50.00.

For equal cardinality N, balanced transport minimizes $\langle P, C\rangle$ over $\mathcal{U}_N = \{P \geq 0 : P\mathbf{1} = \mathbf{1}/N,\ P^\top \mathbf{1} = \mathbf{1}/N\}$; rescaling P by N gives a doubly stochastic matrix whose linear optimum is a permutation, so the optimal total equals the minimum assignment mean (Cuturi, 2013). The entropic objective $\min_{P\in\mathcal{U}_N} \langle P, C\rangle + \varepsilon \sum_{ij} P_{ij}(\log P_{ij} - 1)$ spreads mass onto positive-cost edges for

any positive ε: identical multisets need not yield zero cost, and a row contribution $\sum_j P_{ij}C_{ij}$, $1/N$ times the conditional row cost, is not in the units of Equation 9 without rescaling. For the Sinkhorn comparison in Table 6 we scale the nominal coefficient by a nested median of the normalized matrix and run at most 200 updates with a tolerance, so marginal feasibility is approximate and the effective coefficient can vary slightly across images.

F.3 Features and references

In the original-reference study we score the same 1,568 admitted query–reference pairs and 2,340 distinct image files with both encoders, with feature arrays of shape $2340 \times 256 \times 1024$ in float32. For DINOv2 we take the final output of its 24 blocks. For DINOv3 we take its final normalized patch output after removing the class token and four register tokens. We disable TF32, autocast, compilation, and stochastic behavior, so every score uses single-image, exactly repeatable extraction. Every rule within a configuration receives the same realized cost matrix, and the four configurations share bootstrap weights, normal samples, and category membership.

In the encoder-native study we keep the same test images, both encoders, their input resolutions, and 256 final-layer tokens, and we widen the reference bank to all 1,778 admitted within-category training images (351 breakfast box, 335 juice bottle, 372 pushpins, 360 screw bag, 360 splicing connectors). No query ID or image byte sequence overlaps the bank. To retrieve a reference, we take the float64 mean of the unnormalized patch tokens, L2-normalize the descriptor, and select the most cosine-similar normal image in the query's category, breaking exact ties lexicographically; all rules and both cost normalizations use that reference. On one CPU, our median reduction time per query is about 0.025 ms for NN mean and 1.3–1.8 ms for assignment mean, excluding feature extraction; retrieval adds about 0.3 ms and cost construction about 39 ms.

F.4 Reference construction

For N query tokens and KN pooled reference tokens, our pooled score solves an exact injection from query rows to reference columns with each column used at most once, so spare columns are unpenalized. Reusable matching takes each row minimum on the same matrix. Pooled Chamfer averages the two directional means, and each minimum-over-reference arm takes its own minimum over the same complete reference images.

To see how much the choice of reference matters, we added 117 further independently selected draws (one normal reference per category and draw) to the original three fixed draws, giving the 120-draw sensitivity record. Its logical contrast, averaged over draws, categories and score arms with 2,000 within-draw resamples of the five categories, is $+5.14$ points for DINOv2 and $+4.22$ for DINOv3, as point estimates over the same category populations and images.

For the per-reference bank we use $R \in \{1, 2, 4, 8\}$: the first three references are the fixed draws above, and draws four through eight each add one independently selected reference per category.

G Intervals and statistical scope

G.1 How we computed the intervals

Unless a table says otherwise, our intervals are percentile intervals from paired image bootstraps: each resamples normal, logical, and structural indices separately within category, shares them across every compared arm, and averages category AUROC with equal weight before taking the 2.5th and 97.5th percentiles. Contrasts whose intervals we correct together form a family; for a family, two-sided Bonferroni quantiles replace the pointwise pair, and we state its size. These intervals resample images only, conditioning on fixed scores, references, encoders, and calibration, so they describe these categories rather than a population of categories; analyses that instead resample categories or training seeds are flagged where reported.

Where the gain comes from. For the original references we report pointwise paired 95% intervals from 5,000 image draws for the two strata; the logical stratum split by the dataset's defect subtype

uses 2,000 paired image draws resampled within category. For the encoder-native references with raw costs we use 20,000 paired image draws with Bonferroni quantiles for the four-contrast family.

Pooled, per-reference and matched-reducer banks. The pooled-five intervals are two-sided and adjusted for a family of four. The complete-five column carries pointwise 95% secondary intervals, and one-reference values are point estimates. Both sets of intervals use 20,000 paired image draws. The per-reference bank uses 2,000 paired image bootstraps per R, stratum and arm, pairing pooled-NN with per-reference scores. These intervals are pointwise, and Holm step-down correction covers the four logical p values within each encoder and arm. A second run also resamples the five categories, giving wider, fixed-bank intervals than the image-only ones of Table 10(b). Against the pooled bank, DINOv2 at $R = 2$ is $+7.25$ points $[+1.59, +13.12]$ and DINOv3 at $R = 8$ is $+0.55$ $[-4.47, +5.85]$; for the matched mean reducer, DINOv2 at $R = 2$ is $+6.97$ $[+1.90, +11.72]$ and DINOv3 at $R = 8$ is $+4.54$ $[+1.26, +7.69]$. For the matched reducers, with both arms reduced over the same bank, we use 2,000 paired image bootstraps; the logical intervals are pointwise, and the structural intervals are two-sided 95% with Holm correction across R within each encoder and reducer. Our advance rule counts a structural cost when any interval lies wholly below zero.

Incomplete and contaminated references. Intervals are 20,000 paired image bootstraps sharing resampled indices across every condition and arm, and the control row reproduces the encoder-native row of Table 5(b) to within 3×10^{-14} AUROC points. The control row's intervals there are pointwise, not family-corrected.

Trained detectors. The ten PatchCore decisions form one Bonferroni family. Each lower bound is the one-sided 0.005 quantile of 20,000 paired image draws. A criterion is met only when this exact bound exceeds its margin, so a bound that rounds to zero is not positive. The five EfficientAD-M contrasts form a second family, with one-sided bounds at the 0.01 quantile of 20,000 paired image draws stratified by category and subtype. Logical bounds must exceed zero, and the structural bound must exceed -1 point. The low-FPR endpoint uses 10,000 paired image draws per category and subtype.

Count anomalies beyond LOCO. The four raw CAD-SD contrasts form the primary family. Their 95% intervals come from 20,000 paired draws resampled within dataset and defect type, Bonferroni-adjusted over the four. DigitAnatomy intervals are also paired and subtype-stratified, but pointwise, and no macro average pools the two datasets. The VID-AD lower bounds are one-sided family-four lower bounds from 20,000 paired resamples of the four scenarios. The VisA intervals are paired 95% image-bootstrap intervals.

G.2 What we fixed in advance and what we added later

The single-reference study of Table 5 was not preregistered as a confirmatory test, although its endpoint was frozen. It follows an earlier archived study on the same query–reference pairs, and we fixed the new study's primary contrast after reading those results and before extracting fresh features.

We specified the Real-IAD structural check and the low-FPR family before we saw their outcomes. We added the encoder-native retrieval protocol later and froze its four-contrast family before retrieving or scoring its references. The excess-only comparisons and the CAD-SD reanalysis came later too. We fixed the detector calibration formula and fusion weight, which use no anomaly labels, before computing any detector test score, but after seeing the matching-level test results.

All four pooled intervals include zero, so the pooled logical conjunction is not met. We specified the subtype split after the primary contrast was known, and it uses no new data; the sweep came after the primary detector results.

Two families contain contrasts that were never scored. The low-FPR family had six members, with one-sided bounds at the $0.05/6$ quantile. Its two external ReinAD (Wang et al., 2025) contrasts, on 22 categories, stopped at a normal-image tolerance check in the metal-piece category, before any anomaly outcome was scored. The VID-AD family had four contrasts: two VID-AD quantity contrasts, and two LOCO simple-excess contrasts from a separate detector-training run. We did not

Table 20: **Standing of each analysis.** The subtype split, the neighborhood-size sweep and later comparisons are exploratory.

Contrasts	Standing
PatchCore, ten	Bonferroni family
EfficientAD-M, five	separate Bonferroni family
Encoder-native, four raw-cost matched-mean	primary Bonferroni family
Pooled, four encoder-by-stratum at $K = 5$	Bonferroni family fixed in advance
Category omissions	secondary, pointwise
Encoder-native: pair-median costs, combined anomalies, average precision, excess	secondary
Subtype split, neighborhood-size sweep, later comparisons	exploratory

compute those LOCO contrasts. Both quantity bounds meet the required bound and condition on four scenario units that we had seen before.

G.3 Scope of three secondary analyses

In the sensitivity record over reference draws, every draw reuses the same category populations and images, so these numbers show sensitivity to the choice of reference; they do not estimate a population of references. In the coverage decomposition, we did not measure the missing-versus-added composition of either benchmark's count stratum; the sign change is consistent with that explanation but does not test it. The table shows only that coverage alone does not account for the gain on every benchmark. In the encoder comparison, the ratio of excess to nearest-neighbor cost is computed on normal images, which are shared between the strata of a benchmark, so the rows are not independent measurements: the evidence is two benchmarks by two encoders. Why the two encoders differ remains open.