COUNTED CONDITIONALS: MODELING DISCRETE DISTRIBUTIONS FROM PAIRWISE COUNTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Generative modeling requires estimating high-dimensional distributions from finite data. Counting complete configurations is impractical, but relationships between informative parts recur across examples. We ask how far discrete distributions can be modeled from empirical token-pair counts. We introduce Counted Conditionals, which separates the statistics we store from the way we use them. Rather than pooling observations as separate neighbor votes, we fit tractable joint models on narrow spatial strips from first- and second-order counts. Conditioning sums over unknown tokens within each strip, and a calibrated pool combines the component predictions. Higher-order context counts provide an optional extension. On visual-token and land-cover grids, the strip construction improves completion accuracy and log loss over direct pairwise pooling across three mask patterns. Small-data and increasing-evidence experiments examine how it uses limited training examples and partial observations. We also evaluate the normalized joint induced by applying the pooled predictor in a fixed order, separately from its marginal forecasts. Because records contribute additive statistics, updating the counts and repeating the same deterministic construction reproduces a fresh build under fixed preprocessing and reference information. These results establish a count-based approach to discrete probability modeling and clarify its predictive and computational trade-offs.

1 INTRODUCTION

What does a generative model need to learn about an image? A histogram of colors says little about how a scene is arranged, while counting complete images would require data covering an enormous configuration space. Between these extremes are recurring relationships: which parts appear together, and what observing one part tells us about another. Can these relationships support useful probability models without counting every complete configuration?

The choice of parts matters. Discrete representation learning separates image encoding from modeling token distributions (van den Oord et al., 2017). Masked generators and absorbing discrete diffusion predict missing tokens from observed context (Chang et al., 2022; Li et al., 2023; Ou et al., 2025). These developments motivate our question: *with an informative representation fixed, how far can empirical pairwise relationships take us in constructing conditional distributions?*

Tractable dependence models have long used pairwise relationships (Chow & Liu, 1968; Meilă & Jordan, 2000). Recent work scales probabilistic circuits (Peharz et al., 2020; Liu et al., 2024a) and develops generators for flexible conditional queries (Liu et al., 2025). We revisit explicit co-occurrence statistics in this setting. Recurring pairs can be estimated without enumerating whole neighborhoods, and their counts retain individual training contributions. This is useful when task-specific data are limited or must be revised.

The difficulty is that *pairwise information is not independent evidence*. Nearby observations may repeat the same clue, so multiplying their separate conditional probabilities can overcount evidence. A local voting rule also relies on intermediate guesses to pass information across an extended missing region. These limitations motivate separating the statistics used to represent a distribution from inference within it (Wainwright & Jordan, 2008).

We introduce *Counted Conditionals* (Figure 1). We fit narrow horizontal and vertical strip models from first- and second-order counts. Conditioning sums over unknown tokens within each strip, and a

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.

calibrated pool combines the component predictions. Higher-order context counts provide an optional extension.

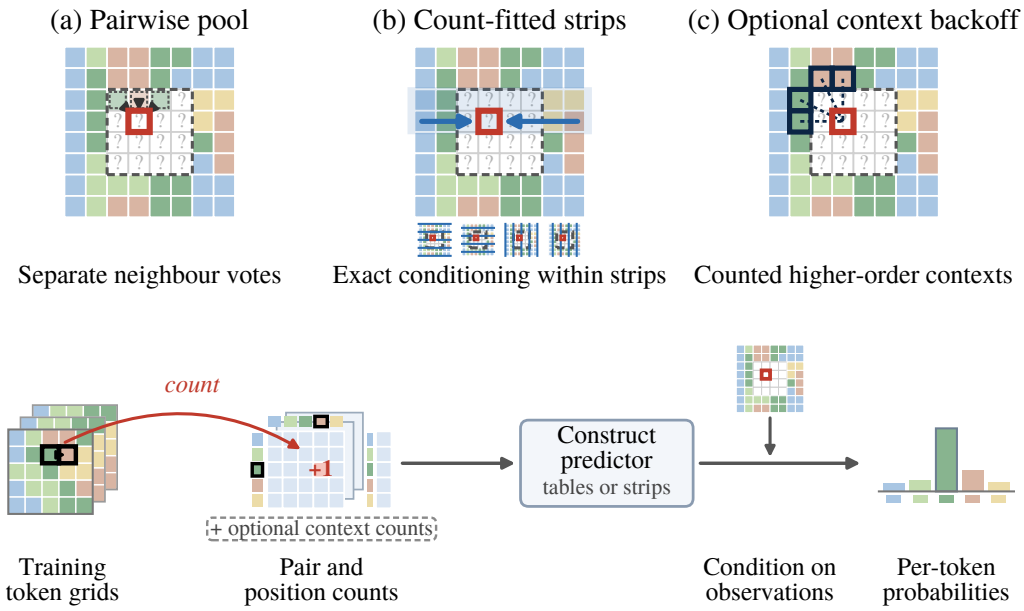


Figure 1: **From recurring token relationships to conditional predictions.** Each observed pair increments a co-occurrence entry. Counts are used to construct a predictor, which is conditioned on a new grid's observations. (a) Pairwise pooling combines separate neighbor scores; interior targets may need earlier fills. (b) Fitted strips sum over unknown intermediate tokens, with exact conditioning within each component. (c) Optional higher-order context counts back off to the strip base.

Contributions. Our contributions are:

1. **A count-fitted probability model.** We combine exact conditioning within tractable strips with calibrated pooling. Applying the pooled predictor in a fixed order also defines a normalized joint for likelihood evaluation.
2. **Evidence on the use of pairwise information.** Strips improve completion accuracy and log loss over direct pooling across three datasets and three masks. Information-theoretic analysis, controlled tests, and small-data and increasing-evidence experiments examine this difference. On NLCD, strip accuracy is within 0.2 percentage points of CNN-S at 8.5 rather than 30.6 kB of predictor state.
3. **Revision through additive statistics.** Following summation-form learning (Cao & Yang, 2015), we update record contributions and reconstruct the predictor. With fixed preprocessing and reference information, deterministic reconstruction matches a fresh build. We measure its quality, time and storage trade-offs.

2 Related Work

Pairwise models and tractable inference. Chow–Liu trees and mixtures of trees construct tractable distributions from pairwise relationships (Chow & Liu, 1968; Meilă & Jordan, 2000). Einsum Networks and PyJuice scale probabilistic-circuit learning and exact inference (Peharz et al., 2020; Liu et al., 2024a). Our bounded-width strip components retain local cycles and use exponential-family sufficient statistics (Wainwright & Jordan, 2008). Composite and piecewise likelihood also exploit tractable subproblems (Varin et al., 2011; Sutton & McCallum, 2005). We fit each component by its own normalized likelihood, then combine conditional marginals through logarithmic pooling (Heskes, 1998), rather than assembling component potentials into a single global model.

Counted spatial contexts. Multiple-point geostatistics uses training-image patterns for conditional simulation: SNESIM stores pattern counts (Strebelle, 2002), while QuickSampling uses efficient pattern matching (Gravey & Mariethoz, 2020). Infini-gram scales count-based prediction with suffix-array lookup and variable-length backoff (Liu et al., 2024b). Our extension instead stores bounded, hashed spatial contexts and backs off to a count-fitted strip predictor. It uses absolute-discount smoothing (Chen & Goodman, 1998), not full Kneser–Ney smoothing (Kneser & Ney, 1995). We distinguish pruned prediction tables from the statistics retained for revision.

Neural conditional generation. Masked generators (Chang et al., 2022; Li et al., 2023), discrete diffusion (Austin et al., 2021; Ou et al., 2025) and random-order autoregression (Pang et al., 2025) learn token distributions. Tracformer explicitly studies generalization to conditioning queries not seen during training (Liu et al., 2025). We study the same conditional-prediction interface through explicit statistics and tractable components. Neural references are adapted to a shared fixed representation, separating predictor comparisons from representation learning.

Data revision and unlearning. Summation-form learning makes training contributions explicit (Cao & Yang, 2015). Deletion-efficient learning, certified removal and SISA address the cost and guarantees of data removal (Ginart et al., 2019; Guo et al., 2020; Bourtoule et al., 2021); UnlearnSPN provides removal for sum–product networks (Becker & Liebig, 2022). Our criterion is agreement with a specified deterministic rebuild under fixed preprocessing and reference information. We evaluate this property alongside prediction quality, update time and maintained state, distinguishing it from accurate but nonmatching warm starts.

3 Constructing Counted Conditionals

Our training collection $\mathcal{D}$ contains N complete categorical grids $x \in \{1, \dots, V\}^{\Omega}$, where Ω indexes grid positions. A query reveals positions $\mathcal{O}$ and asks for the tokens at $\mathcal{U} = \Omega \setminus \mathcal{O}$. The representation is fixed: tokens are quantized image features or native category labels. We construct a pairwise pool as a local reference, fit tractable strip models from first- and second-order counts, and optionally add higher-order contexts.

3.1 Pairwise counts and pooling

For each spatial offset $d \in \Delta_R$, accumulate

$$C_d(u, v) = \sum_{x \in \mathcal{D}} \sum_{i : i, i+d \in \Omega} \mathbf{1}[x_i = u,\, x_{i+d} = v]. \tag{1}$$

Tables are shared across positions with the same offset. R1, R2 and R3 denote square neighbourhoods of radii one, two and three, excluding the target: 8, 24 and 48 offsets. With an add-one estimate $\widehat{p}$ of the token marginal and smoothing strength α, define

$$\widehat{p}_d(u \mid v) = \frac{C_d(u, v) + \alpha \widehat{p}(u)}{\sum_{u'} C_d(u', v) + \alpha}, \qquad e_d(u; v) = \log \widehat{p}_d(u \mid v) - \log \widehat{p}(u). \tag{2}$$

Smoothing shrinks each conditional toward the marginal; the log ratio expresses its evidence relative to that marginal. Let $A_i^{(t)}$ contain the offsets whose neighbours are observed or already filled at round t. We predict

$$q^{(t)}_{\mathrm{pair},i}(u) = \mathrm{softmax}_u \left[b_i(u) + \sum_{d \in A_i^{(t)}} w_d \eta^{(t)}_{i,d} e_d(u; y^{(t)}_{i+d}) \right]. \tag{3}$$

Here $y^{(t)}$ contains observations and earlier predictions; $\eta^{(t)}_{i,d}$ is 1 for an observed neighbour and γ for a predicted one. The prior is $b_i(u) = (1 - \beta) \log \widehat{p}(u) + \beta \log \widehat{p}_i(u)$, where $\widehat{p}_i$ is the position marginal. We use $\beta = 0.5$ on image tokens, $\beta = 0$ for static NLCD, $\gamma = 1$ for the ordinary predictor, and $\alpha = 50$. Offset weights are fitted with 200 Adam steps (Kingma & Ba, 2015) on at most 512 calibration tiles. The pair bank contains $|\Delta_R| V^2$ entries and requires one length-V lookup per available offset and target.

We use either one pass or confidence-based iterative completion (Chang et al., 2022). Our reveal rule distributes the remaining positions across the remaining rounds. Starting with $\mathcal{U}_0 = \mathcal{U}$, at round $t = 0, \ldots, T-1$ select

$$m_t = \left\lceil \frac{|\mathcal{U}_t|}{T-t} \right\rceil, \qquad B_t = \mathrm{Top}_{m_t}\{\max_u q_i^{(t)}(u) : i \in \mathcal{U}_t\}, \tag{4}$$

where Top_{m_t} returns the positions with the largest maximum probabilities. We assign their argmax tokens and set $\mathcal{U}_{t+1} = \mathcal{U}_t \setminus B_t$. Filled tokens provide context in later rounds; original observations remain fixed. The decoder, one pass or $T = 6$ rounds, is selected by hidden-cell accuracy on development data.

3.2 Count-fitted strip models

Recent probabilistic circuits combine learned distributions with tractable inference (Peharz et al., 2020; Liu et al., 2024a). We use fixed pairwise strip graphs. For a horizontal two-row strip, each column state $z_j = (x_{1j}, x_{2j})$ has V^2 values. A component m is a product of disjoint strips, with token-unary and horizontal/vertical pair indicators $\phi_m(x)$ defining an exponential family (Wainwright & Jordan, 2008):

$$q_{\theta,m}(x) = \exp\big(\theta^\top \phi_m(x) - A_m(\theta)\big), \qquad S_{\mathcal{D},m} = \sum_{x \in \mathcal{D}} \phi_m(x). \tag{5}$$

Here A_m is the log-partition function. The average regularized negative log-likelihood is

$$\mathcal{L}_m(\theta) = A_m(\theta) - \theta^\top \frac{S_{\mathcal{D},m}}{N} + \rho\|\theta\|_2^2, \qquad \rho = 10^{-3}. \tag{6}$$

We obtain $\widehat{\theta}_m$ by running L-BFGS (Liu & Nocedal, 1989) from zero with the settings in Appendix C.1. Strip-joint fitting uses only counts; pool calibration additionally uses training tiles.

Forward–backward recursion computes $r_{m,i}(u) = q_{\widehat{\theta}_m,m}(x_i = u \mid x_{\mathcal{O}})$ by summing over unknown tokens within the component. Evidence can pass along a strip without fixing intermediate guesses. The factorized column transition requires two $O(V^3)$ contractions per message, rather than a dense V^4 transition. Appendix B gives the recurrence and enumeration checks.

The four components, H_0, H_1, V_0, V_1, use horizontal and vertical strips at two alignments. Shifted alignments disable coupling across the rolled boundary, avoiding periodic adjacency. Image models use position-specific unaries; NLCD shares them. We combine the conditioned marginals using a log-linear pool related to logarithmic opinion pooling (Heskes, 1998):

$$q_{\mathrm{strip},i}(u) = \mathrm{softmax}_u\left[a_0 \log \pi_i(u) + \sum_{m=1}^{4} a_m \log r_{m,i}(u)\right], \tag{7}$$

Here π_i is the add-one position marginal for images or the add-one shared marginal for NLCD. It is strictly positive (Appendix C.1). We fit the five coefficients by masked cross-entropy on up to 512 training tiles, restricted to the available subset in small-data fits. The unconstrained prior coefficient is approximately -0.2 on image tasks and -0.30 on NLCD, with approximately 0.3 per strip expert. Conditioning is exact within each component; Eq. 7 produces a normalized per-token pool. Appendix X evaluates both the component joints and the joint induced by using the pool in a fixed order. Under six-round decoding, earlier predictions are clamped as evidence while original observations remain fixed.

3.3 Higher-order context backoff

Multiple-point geostatistics conditions on spatial patterns (Strebelle, 2002; Gravey & Mariethoz, 2020), while recent count-based language models use variable-length contexts and backoff (Liu et al., 2024b). Our extension counts combinations of up to K nearest genuinely observed tokens within radius R, ordered by a fixed rule and paired with their offsets. Keys map to a fixed number of hash buckets; collisions are part of the predictor.

We estimate target-token distributions with absolute-discount backoff through shorter contexts to a base predictor, adapting count-smoothing methods from language modeling (Chen & Goodman, 1998). Writing the resulting distribution as h_i, we combine it with the base:

$$q_{\mathrm{ctx},i}(u) \propto h_i(u)^{\lambda} q_{\mathrm{base},i}(u)^{1-\lambda}. \tag{8}$$

The base is the pairwise or strip pool. Our default extension uses R3 K4, prediction-view count threshold 25 and $\lambda = 0.3$ on 16×16 image grids and 32×32 maps. Context keys never include generated tokens. Unlike strip-only prediction, this extension retains higher-order empirical patterns.

Record-level context masks follow a fixed deterministic rule to preserve additive updates (Cao & Yang, 2015). We either retain unpruned bucket statistics and reconstruct the pruned prediction view, or fix a dictionary using independent reference data. Unpruned statistics matter because edits can move a context across the retention threshold. The low-memory CIFAR variant freezes keys with at least ten occurrences in an independent reference collection, then fits target-data counts for those keys only. This rule is selected on development data; the full K4 bank remains the default. A fixed dictionary reduces both maintained statistics and prediction storage, whereas pruning only the prediction view need not reduce the maintained statistics.

4 What does pairwise pooling assume?

The founding hypothesis concerns the usefulness of pairwise information. A particular pooling rule adds an additional assumption. We make that assumption explicit for the unweighted population version of Eq. 3. Let X be the target and $Y = (Y_1, \ldots, Y_k)$ the observed context under a distribution P, with positive probabilities on the relevant support. The marginal-corrected pool is the conditional of the star model

$$R(x, y) = P(x) \prod_{j=1}^{k} P(y_j \mid x), \qquad q_\star(x \mid y) = R(x, y)/R_Y(y). \tag{9}$$

Bayes' rule makes this the product of $P(x \mid y_j)/P(x)$ with one copy of $P(x)$.

Proposition 1 (Population pooling error)**.** *For Eq. 9,*

$$\mathbb{E}_{P(Y)} \mathrm{KL}(P(X \mid Y) \| q_\star(X \mid Y)) = \mathrm{TC}(Y \mid X) - \mathrm{KL}(P(Y) \| R_Y(Y)) \,. \tag{10}$$

The identity follows by expanding $\mathrm{KL}(P(X, Y) \| R(X, Y))$ in two orders. It separates the conditional dependence among neighbours from a cancellation through the evidence distribution. Pairwise correlation strength alone does not determine the error. Conditional independence of neighbours given the target makes the pool exact; a general pairwise grid does not imply this star-shaped factorization.

Proposition 2 (Information omitted by a context)**.** *For all observed information H, a fixed selected subset C, and any predictor $q(X \mid C)$,*

$$\mathbb{E}_H \mathrm{KL}(P(X \mid H) \| q(X \mid C)) = I(X; H \setminus C \mid C) + \mathbb{E}_C \mathrm{KL}(P(X \mid C) \| q(X \mid C)). \tag{11}$$

The first term measures information that the predictor does not use. The second measures its error on the context it does use. Adding observations can lower the first term yet worsen a particular predictor through the second. This explains why expanding a bank of pairwise votes is not the same as improving conditioning. The controlled experiment in Appendix M quantifies this effect for pooling. The proofs and the signed finite-sample correction are in Appendix A.

5 Updating an explicit statistical representation

Each model stores additive statistics $S_{\mathcal{D}} = \sum_{x \in \mathcal{D}} \phi(x)$, including the record count N. The representation, offsets, hash function, dictionary and contribution-mask rule define ϕ. Removing records R and adding records B gives

$$S_{\mathcal{D}'} = S_{\mathcal{D}} - \sum_{x \in R} \phi(x) + \sum_{x \in B} \phi(x). \tag{12}$$

A correction subtracts the old record's contribution and adds the new one. Lookup probabilities, strip-joint parameters and context views are reconstructed from the revised statistics.

The full strip predictor additionally uses calibration inputs $\mathcal{C}_{\mathcal{D}}$, drawn reproducibly from the training collection. Write its construction as $F(S_{\mathcal{D}}, \mathcal{C}_{\mathcal{D}}; \mathcal{R})$, with fixed preprocessing and reference information $\mathcal{R}$. Updating $S_{\mathcal{D}'}$ and regenerating $\mathcal{C}_{\mathcal{D}'}$ by the same rule reproduces the corresponding fresh-build inputs. The same deterministic fitting and selection procedure then produces the same predictor. Finite-iteration fitting is a valid reference; a unique optimum is not required. Data-dependent stages must be repeated or fixed independently of removable records. Strip-pool recalibration reads 512 retained training records, so the full rebuild is not statistics-only (Appendix F).

For overlapping map windows, we update only pairs with an edited endpoint; Appendix N gives the algorithm and its per-window bound.

6 Experiments

6.1 Experimental setup

We evaluate completion on CIFAR-10 (Krizhevsky, 2009), ImageNet (Deng et al., 2009) and NLCD (U.S. Geological Survey, 2024). Image tasks use fixed DINOv2 features (Oquab et al., 2023), which we cluster into 16 categories on 16×16 grids; NLCD uses 16 labels on 32×32 windows at 240 m mode aggregation. Central, off-centre and scattered masks each hide 25% of cells. Images are encoded before token masking: this is categorical-grid completion, not RGB inpainting (Appendix U).

Table 1: **Central-hole completion on common test grids.** Absolute hidden-token accuracy (%) and calibrated marginal NLL (nats) on 3,000 tiles per dataset. Each hole hides 25% of cells. Parentheses denote accuracy standard deviations over three training seeds; other rows are individual fits.

	CIFAR-10		ImageNet		NLCD	
Predictor	Acc.↑	NLL↓	Acc.↑	NLL↓	Acc.↑	NLL↓
Local mode (5×5)	72.03	0.967	63.56	1.273	73.83	1.088
Pairwise	84.54	0.587	72.70	1.052	73.51	1.042
Fitted local	84.84 (0.05)	0.572	72.98 (0.03)	1.038	73.13 (0.00)	1.077
Strips	85.20	0.474	73.41	0.889	74.30	0.957
Strips + K4 contexts	85.79	0.458	74.35	0.887	74.74	0.926
Strips + fixed dictionary	85.70	0.463	—	—	—	—
CNN-S	86.42 (0.05)	0.410	74.97 (0.02)	0.761	74.33 (0.14)	0.868
MaskGIT-tiny	85.90	0.422	74.14	0.771	74.56	0.810
RADD	86.28	0.400	74.26	0.799	75.00	0.704
MAGE	86.99	0.409	75.44	0.773	75.28	0.731
MaskGIT-medium	87.35	0.353	76.30	0.683	75.32	0.693

Pairwise uses R1 on images and R3 on NLCD. Every row uses its development-selected decoder and temperature. The full K4 bank is the default; the fixed dictionary is a separate CIFAR variant. Model references: MaskGIT (Chang et al., 2022), RADD (Ou et al., 2025) and MAGE (Li et al., 2023). All scores are from our token-grid evaluation. Complete three-mask results and dataset-specific model state are in Tables 3–5.

Every method within a dataset uses the same 3,000 test tiles. Count-based models use 50,000 CIFAR training tiles and 200,000 on ImageNet and NLCD, matching CNN-S. Decoder and temperature are selected on development data. References include a 5×5 local vote, the fitted additive local control and CNN-S, alongside task-specific implementations of MaskGIT (Chang et al., 2022), RADD (Ou et al., 2025) and MAGE (Li et al., 2023). These model families are evaluated on the shared categorical representation; the reported scores are measured here rather than taken from their original papers.

Table 1 reports absolute central-hole accuracy and calibrated marginal NLL; Tables 3–5 give all masks and state. CNN-S and the control show three-seed means and accuracy standard deviations. Other

rows are individual fits. Predictor state excludes the shared encoder/codebook, working memory and maintained statistics. Protocols and paired comparisons are in Appendices C and E.

6.2 Completion results

Strips improve accuracy and calibrated NLL over pairwise pooling on every dataset and mask. CIFAR central accuracy rises from 84.54% to 85.20%, with NLL falling from 0.587 to 0.474; off-centre accuracy rises from 73.64% to 76.28%. ImageNet strips reach 73.41% and 65.77% on the two holes, versus 72.70% and 64.25% for pooling. These gains precede the higher-order extension.

The land-cover strip model uses 8.5 kB, about 28% of CNN-S's 30.6 kB prediction state. Central accuracy is 74.30% versus 74.33%; off-centre and scattered accuracies are each 0.19 points higher (74.41% and 84.32%). Strips exceed both pairwise R3 and the local-vote means on every mask. CNN-S retains lower NLL. Figure 2 illustrates spatial improvements and remaining errors.

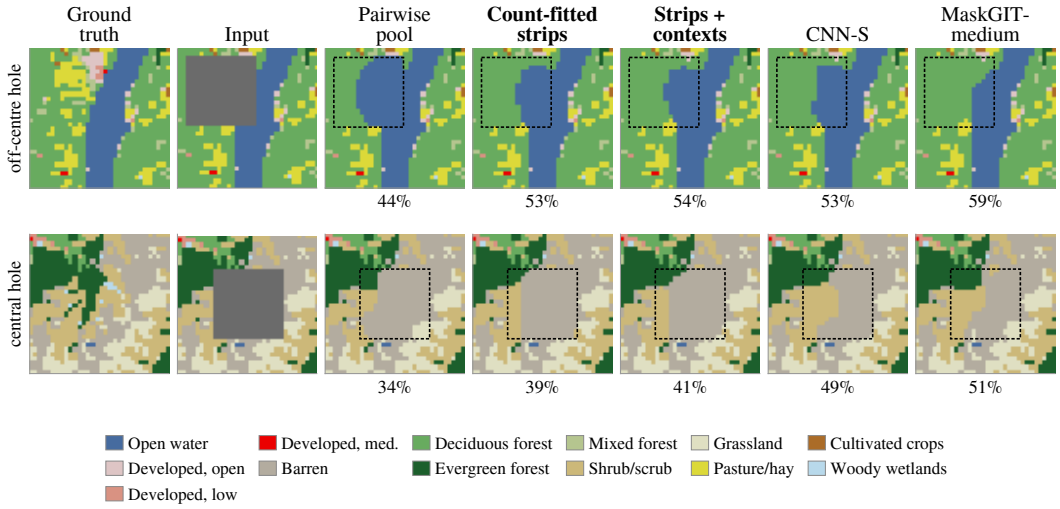


Figure 2: **Land-cover completion.** Gray cells are hidden; percentages measure accuracy only there. The upper example shows improvement over pairwise pooling, while the lower retains a gap to the neural references. These are illustrative selections from an outcome-stratified export, not representative frequency estimates. CNN-S uses seed 42. Appendix L gives the selection rule and CIFAR examples.

6.3 Higher-order contexts and storage

K4 contexts raise central accuracy to 85.79% on CIFAR, 74.35% on ImageNet and 74.74% on NLCD, with little added scattered-mask accuracy. On CIFAR's off-centre hole, accuracy improves but NLL worsens. The banks require 5.4, 24.6 and 55.8 MB, respectively. The NLCD extension exceeds the CNN-S accuracy means on all masks, at much greater state. Stronger neural references remain ahead: CIFAR central accuracy is 86.28% for RADD and 87.35% for MaskGIT-medium.

The separate reference-frozen CIFAR dictionary retains 85.70% central accuracy at 0.68 MB of prediction state and 1.3 MB of maintained statistics, versus 5.4 and 126.5 MB for the full bank. Its scattered accuracy is below strips alone (83.59% versus 83.85%). Appendix P gives the selection rule and complete storage ablation.

6.4 Fitting from limited task data

Figure 3 tests three draws of each task-training subset, with count and aggregation fits restricted to that subset. At 100 tiles, strips reach 84.39% central accuracy and NLL 0.528, compared with 81.37% and 0.661 for CNN-S; contexts give 84.72% and 0.518. The fitted local control has the highest central accuracy at this budget, while strips improve its NLL and scattered accuracy. At 300 tiles, strips

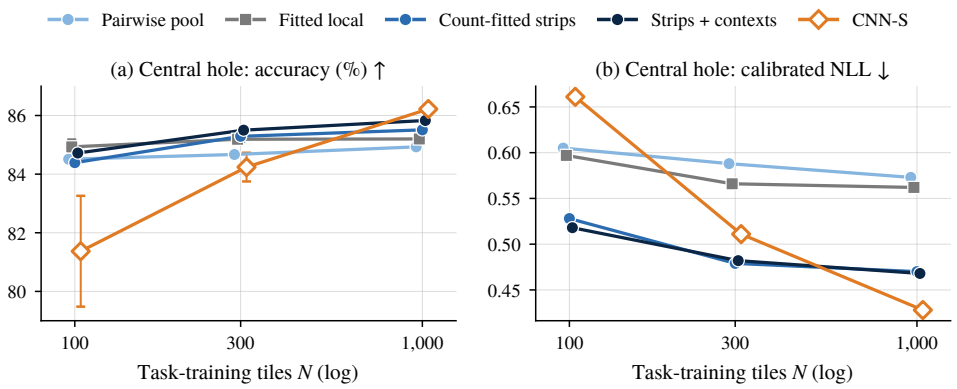


Figure 3: **Fitting from limited task data.** Central-hole predictions on 1,500 CIFAR development tiles. Count and aggregation fits use only N task-training tiles; the pretrained representation is fixed. Accuracy bars are half-ranges over three subset draws, not confidence intervals; NLL curves show means. Both masks and all values are in Table 24.

exceed the CNN means on both masks. At 1,000, CNN-S leads on the hole and scattered accuracies are close (Table 24). The fitted joint construction retains useful small-data behavior at these budgets.

6.5 Using additional observations

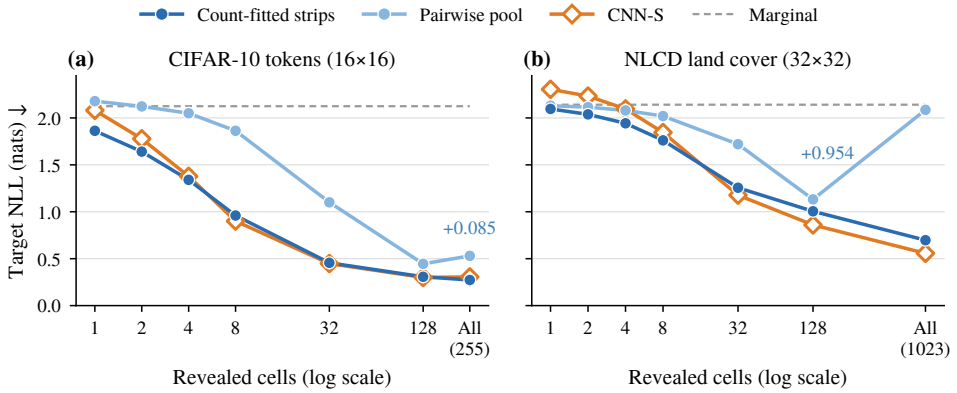


Figure 4: **More observations do not necessarily improve pairwise pooling.** We hold one hidden target fixed in each of 500 development grids and reveal nested sets of genuine observations. No predicted tokens are used as evidence. Strip NLL decreases across the tested levels; the pool's NLL increases after the final reveal. Complete values are in Appendix Q.

Recent work on flexible conditional generation also examines generalization across conditioning queries (Liu et al., 2025). Here, Figure 4 fixes one hidden target in each of 500 development grids and reveals nested sets of genuine observations. With one revealed cell, CIFAR strip NLL is 1.863, versus 2.179 for pooling and 2.081 for CNN-S; the NLCD values are 2.096, 2.130 and 2.305. As evidence increases, strip NLL falls to 0.273 and 0.697. The pool initially improves, but its final reveal raises NLL from 0.445 to 0.530 on CIFAR and 1.132 to 2.086 on NLCD. Thus adding observations can hurt the pooling rule while improving strip predictions, consistent with the distinction in Section 4. Appendices Q and R report all values and distance breakdowns.

Joint likelihood. On the same 3,000 test tiles per dataset, fixed-raster teacher forcing gives the strip pool lower conditional joint NLL than the proper strip mixture in all nine settings, including at temperature one. CNN-S and both MaskGIT references have lower ordered losses. Appendix X reports the complete comparison and distinguishes joint likelihood from marginal losses.

6.6 REVISION QUALITY, TIME AND STORAGE

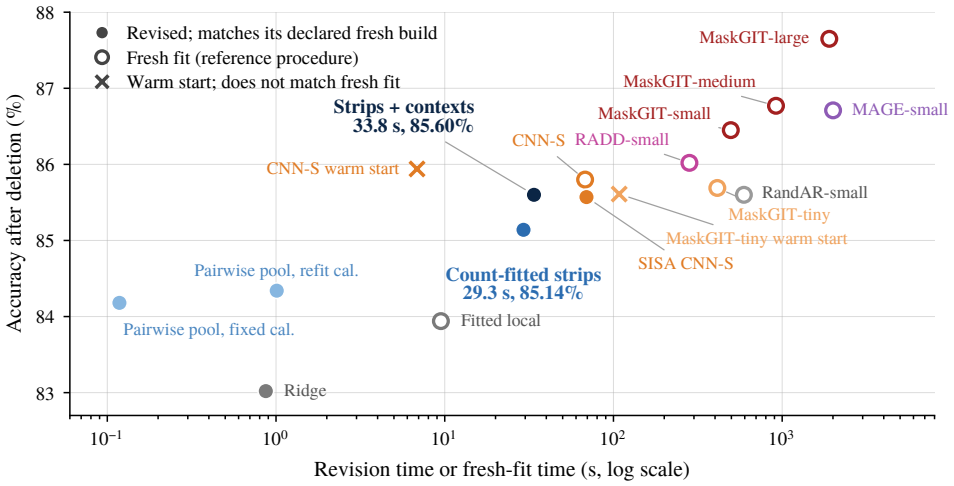


Figure 5: **Quality and cost after deleting 4,500 of 45,000 CIFAR records.** All procedures use the same 2,000-tile cohort. Filled markers pass the reported rebuild checks; open markers are fresh fits; crosses are nonmatching warm starts. SISA uses four shards and four slices. Strip times exclude decoder/temperature selection; their statistics exclude calibration records. Table 9 and Appendix F give the timing and state boundaries.

Figure 5 compares count-based updates with neural refits, warm starts and sliced SISA (Bourtoule et al., 2021) on the same 2,000-tile CIFAR deletion cohort. Strips reach 85.14% accuracy and NLL 0.481 in 29.3 s; contexts give 85.60% and 0.465 in 33.8 s. These update-and-refit times exclude decoder and temperature selection. The statistic inventories are 0.1 MB and 121.6 MB; pool recalibration additionally reads 512 retained training records. Both pass the fresh-build checks in Appendix F (Table 9).

Fixed-calibration pairwise pooling revises in 0.118 s at 84.18% accuracy. CNN-S reaches 85.80% after a 67.91 s fresh fit; larger neural models gain accuracy at greater measured fitting cost. Warm starts are fast and accurate but do not reproduce the specified fresh fit. Ridge and sliced SISA pass their own rebuild checks. Appendix F gives the complete procedures and state accounting.

7 DISCUSSION AND CONCLUSION

Relationships between informative parts offer a way to model discrete distributions without enumerating complete configurations. Counted Conditionals separates stored statistics from their use in prediction. Strips improve on pairwise votes, and context backoff adds higher-order patterns. In held-out likelihood evaluation, the raster-ordered strip pool outperforms the proper component mixture on all nine dataset–mask settings, even without temperature scaling (Appendix X). Small-data and evidence experiments show how the construction uses limited data and observations; additive counts preserve its revision path.

At $V = 64$, strips improve all three test masks over pairwise pooling; the CNN retains the central-hole advantage, while strips lead that checkpoint off-centre and on scattered masks (Appendix J). Component calibration, larger token parts and neural continuation are examined in Appendices I, S and K.

Scope. The experiments use fixed categorical grids. Image features are computed before token masking; encoder pretraining is outside the reported predictor costs and deletion scope. Conditioning is exact within each strip component, while the calibrated pool combines their conditional marginals. The sampling pilot evaluates its ordered generator separately. Neural references often achieve lower NLL, and context banks can be large. Training-seed variation is measured for CNN-S and the fitted control. Pairwise storage grows with V^2 and strip inference with V^3 per message. Representation choice and coherent sampling remain directions for the broader pairwise hypothesis.

AI USE STATEMENT

Generative AI tools assisted with literature discovery, research ideation and execution, implementation, result interpretation, and drafting and polishing text, under the authors' supervision. This included exploring experimental directions, planning comparisons and ablations, and assisting with code development and debugging. The authors conducted the experiments; all reported results come from actual experimental runs, not AI-generated estimates. The authors reviewed and revised all AI-assisted material and take full responsibility for the methods, results, citations, and final manuscript.

ETHICS STATEMENT

The experiments concern research benchmarks and categorical land-cover completion, not deployed decisions about people or land use. Predicted labels are not observations and should not be treated as ground truth. The image experiments inherit the content and limitations of the source datasets and pretrained representation. Agreement with a fresh build after record edits is not a general privacy or legal-compliance guarantee; the specified fixed reference data and preprocessing remain outside the deletion scope.

REPRODUCIBILITY STATEMENT

Appendix A provides proofs and Appendix B the strip recurrence and numerical checks. Appendices C and F specify the data roles, evaluation sets and uncertainty conventions. The remaining appendices report broader model comparisons, ablations, context-state inventories and coding results. They distinguish prediction payload, maintained statistics, calibration dependencies and timing boundaries for the evaluated procedures. The code will be released publicly upon publication.

REFERENCES

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured Denoising Diffusion Models in Discrete State-Spaces. In *Advances in Neural Information Processing Systems*, 2021. URL https://arxiv.org/abs/2107.03006.

Alexander Becker and Thomas Liebig. Certified Data Removal in Sum-Product Networks. *arXiv preprint arXiv:2210.01451*, 2022. URL https://arxiv.org/abs/2210.01451.

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine Unlearning. In *IEEE Symposium on Security and Privacy*, 2021. URL https://arxiv.org/abs/1912.03817.

Yinzhi Cao and Junfeng Yang. Towards Making Systems Forget with Machine Unlearning. In *IEEE Symposium on Security and Privacy*, pp. 463–480, 2015. URL https://doi.org/10.1109/SP.2015.35.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. MaskGIT: Masked Generative Image Transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11315–11325, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Chang_MaskGIT_Masked_Generative_Image_Transformer_CVPR_2022_paper.html.

Stanley F. Chen and Joshua Goodman. An empirical study of smoothing techniques for language modeling. Technical Report TR-10-98, Computer Science Group, Harvard University, August 1998. URL https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/tr-10-98.pdf.

C. K. Chow and C. N. Liu. Approximating Discrete Probability Distributions with Dependence Trees. *IEEE Transactions on Information Theory*, 14(3):462–467, 1968. URL https://doi.org/10.1109/TIT.1968.1054142.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009. URL https://www.image-net.org/about.

Antonio Ginart, Melody Guan, Gregory Valiant, and James Zou. Making AI Forget You: Data Deletion in Machine Learning. In *Advances in Neural Information Processing Systems*, 2019. URL https://arxiv.org/abs/1907.05012.

Mathieu Gravey and Grégoire Mariethoz. QuickSampling v1.0: A robust and simplified pixel-based multiple-point simulation approach. *Geoscientific Model Development*, 13(6):2611–2630, 2020. doi: 10.5194/gmd-13-2611-2020. URL https://gmd.copernicus.org/articles/13/2611/2020/.

Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens van der Maaten. Certified Data Removal from Machine Learning Models. In *Proceedings of the International Conference on Machine Learning*, pp. 3832–3842, 2020. URL https://proceedings.mlr.press/v119/guo20c.html.

Tom Heskes. Selecting Weighting Factors in Logarithmic Opinion Pools. In *Advances in Neural Information Processing Systems 10*, pp. 266–272, 1998. URL https://papers.neurips.cc/paper/1413-selecting-weighting-factors-in-logarithmic-opinion-pools.pdf.

Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1412.6980.

Reinhard Kneser and Hermann Ney. Improved Backing-off for M-gram Language Modeling. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 181–184, 1995. URL https://doi.org/10.1109/ICASSP.1995.479394.

Alex Krizhevsky. Learning Multiple Layers of Features from Tiny Images. Technical report, University of Toronto, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Tianhong Li, Huiwen Chang, Shlok Kumar Mishra, Han Zhang, Dina Katabi, and Dilip Krishnan. MAGE: MAsked Generative Encoder to Unify Representation Learning and Image Synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. URL https://arxiv.org/abs/2211.09117.

Anji Liu, Kareem Ahmed, and Guy Van den Broeck. Scaling tractable probabilistic circuits: A systems perspective. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 30630–30646. PMLR, 2024a. URL https://proceedings.mlr.press/v235/liu24a.html.

Anji Liu, Xuejie Liu, Dayuan Zhao, Mathias Niepert, Yitao Liang, and Guy Van den Broeck. Tractable transformers for flexible conditional generation. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 38963–38986. PMLR, 2025. URL https://proceedings.mlr.press/v267/liu25ao.html.

Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45:503–528, 1989. doi: 10.1007/BF01589116. URL https://doi.org/10.1007/BF01589116.

Jiacheng Liu, Sewon Min, Luke Zettlemoyer, Yejin Choi, and Hannaneh Hajishirzi. Infini-gram: Scaling unbounded n-gram language models to a trillion tokens. In *First Conference on Language Modeling*, 2024b. URL https://openreview.net/forum?id=u2vAyMeLMm.

Marina Meilă and Michael I. Jordan. Learning with Mixtures of Trees. *Journal of Machine Learning Research*, 1:1–48, 2000. URL https://jmlr.org/papers/v1/meila00a.html.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, et al. DINOv2: Learning Robust Visual Features without Supervision. *arXiv preprint arXiv:2304.07193*, 2023. URL https://arxiv.org/abs/2304.07193.

Jingyang Ou, Shen Nie, Kaiwen Xue, Fengqi Zhu, Jiacheng Sun, Zhenguo Li, and Chongxuan Li. Your Absorbing Discrete Diffusion Secretly Models the Conditional Distributions of Clean Data. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=0dcMmg0v1l.

Ziqi Pang, Tianyuan Zhang, Fujun Luan, Yunze Man, Hao Tan, Kai Zhang, William T. Freeman, and Yu-Xiong Wang. RandAR: Decoder-only Autoregressive Visual Generation in Random Orders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 45–55, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Pang_RandAR_Decoder-only_Autoregressive_Visual_Generation_in_Random_Orders_CVPR_2025_paper.html.

Robert Peharz, Steven Lang, Antonio Vergari, Karl Stelzner, Alejandro Molina, Martin Trapp, Guy Van den Broeck, Kristian Kersting, and Zoubin Ghahramani. Einsum networks: Fast and scalable learning of tractable probabilistic circuits. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 7563–7574. PMLR, 2020. URL https://proceedings.mlr.press/v119/peharz20a.html.

Sebastien Strebelle. Conditional Simulation of Complex Geological Structures Using Multiple-Point Statistics. *Mathematical Geology*, 34(1):1–21, 2002. URL https://doi.org/10.1023/A:1014009426274.

Charles Sutton and Andrew McCallum. Piecewise Training for Undirected Models. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pp. 568–575, 2005. URL https://arxiv.org/abs/1207.1409.

U.S. Geological Survey. Annual National Land Cover Database. Dataset programme and product documentation, 2024. URL https://www.usgs.gov/centers/eros/science/annual-national-land-cover-database.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural Discrete Representation Learning. In *Advances in Neural Information Processing Systems*, 2017. URL https://arxiv.org/abs/1711.00937.

Cristiano Varin, Nancy Reid, and David Firth. An Overview of Composite Likelihood Methods. *Statistica Sinica*, 21:5–42, 2011. URL https://www3.stat.sinica.edu.tw/sstest/j21n1/j21n11/j21n11.html.

Martin J. Wainwright and Michael I. Jordan. Graphical Models, Exponential Families, and Variational Inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305, 2008. URL https://doi.org/10.1561/2200000001.

A PROOFS AND SCOPE OF THE IDENTITIES

A.1 POPULATION POOLING

By Bayes' rule, $P(x \mid y_j)/P(x) = P(y_j \mid x)/P(y_j)$. Terms $P(y_j)$ do not depend on x, hence marginal-corrected pooling normalizes the same score as $R(x, y) = P(x) \prod_j P(y_j \mid x)$. Because each factor is normalized conditional on x, R is a normalized joint with $R_X = P_X$.

Conditional total correlation is

$$\mathrm{TC}(Y \mid X) = \mathbb{E}_{P(X)} \mathrm{KL}\left(P(Y \mid X) \middle\| \prod_j P(Y_j \mid X) \right). \tag{13}$$

Consequently,

$$\mathrm{KL}(P(X,Y) \| R(X,Y)) = \mathbb{E}_{P(X,Y)} \log \frac{P(Y \mid X)}{\prod_j P(Y_j \mid X)} \tag{14}$$

$$= \mathrm{TC}(Y \mid X). \tag{15}$$

Expanding the same joint divergence by conditioning on Y gives

$$\mathrm{KL}(P(X,Y) \| R(X,Y)) = \mathrm{KL}(P(Y) \| R_Y(Y)) + \mathbb{E}_{P(Y)} \mathrm{KL}(P(X \mid Y) \| R(X \mid Y)). \tag{16}$$

Subtracting proves Proposition 1. This is an application of the KL chain rule, not a new divergence. In particular, $\mathrm{TC}(Y \mid X)$ alone is not the pooling error: the evidence divergence must also be subtracted. No monotonicity with the number or radius of neighbours follows from this identity.

A.2 OMITTED OBSERVATIONS AND EMPIRICAL ESTIMATION

For a fixed observation pattern and subset C of the observed information H, add and subtract $\log P(X \mid C)$:

$$\mathbb{E}_{P(X,H)} \log \frac{P(X \mid H)}{q(X \mid C)} = \mathbb{E}_{P(X,H)} \log \frac{P(X \mid H)}{P(X \mid C)} + \mathbb{E}_{P(X,C)} \log \frac{P(X \mid C)}{q(X \mid C)} \tag{17}$$

$$= I(X; H \setminus C \mid C) + \mathbb{E}_{P(C)} \mathrm{KL}(P(X \mid C) \| q(X \mid C)). \tag{18}$$

This proves Proposition 2. Random mask or context indices must be included in the conditioning information, or the identity applied separately conditional on them. It must not be used to equate a generated-history loss with a loss conditional only on genuine observations.

For an empirical predictor $\widehat{q}$ and its designated population pooled counterpart $q_\star$, a further exact rearrangement is

$$\mathbb{E}_H \mathrm{KL}(P(X \mid H) \| \widehat{q}(X \mid C)) = I(X; H \setminus C \mid C) + \mathbb{E}_C \mathrm{KL}(P(X \mid C) \| q_\star(X \mid C)) \tag{19}$$

$$+ \mathbb{E}_{P(X,C)} \log \frac{q_\star(X \mid C)}{\widehat{q}(X \mid C)}. \tag{20}$$

The last term is signed for a realized training sample. It is not generally an independent nonnegative estimation penalty. Temperature fitting, shrinkage, weight fitting and the choice of $q_\star$ all affect its interpretation. The controlled experiment uses the stated unweighted construction and genuine context, with shrinkage strength 50.

A.3 FRESH-BUILD AGREEMENT

Fix the representation, reference information $\mathcal{R}$, seed rules and deterministic construction F. Additivity gives $S_{\mathcal{D}} - \sum_{r \in R} \phi(r) + \sum_{b \in B} \phi(b) = S_{\mathcal{D}'}$ exactly in integer arithmetic, subject to sufficient integer range. For the full strip predictor, both reconstruction routes also use the same calibration inputs $\mathcal{C}_{\mathcal{D}'}$, selected by the same rule from the same revised collection. Applying $F(S_{\mathcal{D}'}, \mathcal{C}_{\mathcal{D}'}; \mathcal{R})$ to these equal inputs yields equal outputs. Equality of counts alone is not sufficient when calibration inputs depend on retained records. This statement does not require a unique optimum: finite-iteration

deterministic fitting is the reference procedure. A warm start is a different procedure unless its output is checked to agree.

In a context model, a record may contribute under several masks. Those masks must be fixed by the declared reproducible rule; a fresh random draw would change $\phi(r)$. Repeated records contribute with their multiplicity. For a correction whose seed depends on content, subtract the old record's old masks and add the new record's new masks. If a learned tokenizer, dictionary, prior or calibration procedure depends on the removable data, its change must also be included. The statement is not an end-to-end guarantee for a tokenizer pretrained on overlapping data.

B STRIP MODELS AND EXACT CONDITIONING

For a two-row strip of length L, let $z_j = (u_j, v_j)$. A concrete pairwise energy can be written with within-column term U_j and horizontal pair terms H_1, H_2:

$$q_\theta(z_{1:L}) \propto \prod_{j=1}^{L} \exp U_j(u_j, v_j) \prod_{j=2}^{L} \exp\{H_1(u_{j-1}, u_j) + H_2(v_{j-1}, v_j)\}. \tag{21}$$

The term U_j contains the two token unaries and their vertical coupling. To condition on observations, multiply the within-column factor by indicators fixing the observed coordinates. The forward message is

$$f_j(u, v) = \psi_j(u, v) \sum_{u', v'} f_{j-1}(u', v') \exp H_1(u', u) \exp H_2(v', v), \tag{22}$$

where ψ_j includes clamping. The two horizontal factors allow successive summation over u' and v' in two $O(V^3)$ contractions. Backward messages use the reversed recursion. Normalized products of forward and backward messages give the column-state posterior; summing one coordinate gives each token marginal. Implementations operate in log space.

The partition function with no clamping is Z; with observed tokens clamped it is $Z_{\mathcal{O}}$. Thus $q_\theta(x_{\mathcal{O}}) = Z_{\mathcal{O}}/Z$. A component over disjoint strips multiplies these strip distributions. Horizontal and vertical components can be combined as a proper joint mixture, with posterior component weights proportional to $\pi_m q_m(x_{\mathcal{O}})$. This mixture is a different construction from the conditional product in Eq. 7.

Numerical checks. On a 2×4 strip with $V = 3$ (6,561 states), enumeration and the recursion agree within 1.8×10^{-15} on the log partition and observed-data log likelihood, and within 6.4×10^{-8} on stored float32 conditional marginals. Automatic and finite-difference gradients differ by at most 2.2×10^{-10}; sufficient-statistic and explicit-record likelihoods differ by at most 2.3×10^{-13}. The shifted-strip test on an enumerated binary 4×4 grid reports zero log-partition/observed-likelihood discrepancy and 7.8×10^{-8} marginal discrepancy. These checks compare normalization, conditioning and gradients against exact finite enumeration.

C EVALUATION PROTOCOL

Representation. CIFAR and ImageNet use fixed DINOv2 features clustered into 16 categories on 16×16 grids. NLCD uses 16 land-cover labels on 32×32 windows at 240 m mode aggregation, with stride 16. Image-source identities separate image splits; geographic cells separate map splits. The query hides already encoded tokens. An observed DINOv2 token can therefore contain information about other image positions, so the task is not equivalent to masking RGB pixels before an encoder. Every image-token predictor receives the same token representation.

Training budgets. The principal count-based models use 50,000 task-training tiles on CIFAR and 200,000 on ImageNet and NLCD, matching the CNN-S training budgets. These counts do not include encoder pretraining. Strip-pool calibration uses up to 512 task-training tiles; decoder and post-hoc temperature selection use development examples. In the small-data study, counts and aggregation fits use only the stated subset. The CIFAR fixed dictionary is selected on an independent reference collection and is not part of that small-data comparison.

Evaluation sets. Tables 3–5 use 3,000 common test tiles per dataset. The CIFAR set was previously used by the frozen-hybrid experiment and is separate from the subsequent model-development and

selection examples; it is not claimed to have been unseen throughout the entire project. ImageNet uses test-partition images and NLCD uses windows from geographic evaluation cells. The earlier 2,000-tile comparisons in Appendix H are a separate cohort. No absolute score is transferred between these cohorts.

Masks and selection. Central holes are 8×8 on image-token grids and 16×16 on maps. The off-centre image hole occupies indices 1–8 in each coordinate. Scattered masks in the principal comparison hide 25%; the earlier NLCD comparison also includes a separate random-50% protocol. One pass or six confidence rounds and one probability temperature are selected on development tiles for each evaluated row. The architecture and model hyperparameters are fixed before test scoring. A post-hoc scalar temperature is applied to saved scores without changing the completed trajectory.

Small-data fitting. Each budget uses three subset draws. Strip-pool and pairwise calibration use $\min(512, N)$ task-training tiles; the context threshold is $\max(2, \min(25, N/20))$. CNN-S uses 12,000 steps: its first two runs reuse checkpoints from the same subset rule, selected on the disjoint development slice; the third uses the same training recipe. Table 24 reports accuracy half-ranges over the three draws, not standard deviations. It is a development study, distinct from the 3,000-tile prediction tests.

Metrics and uncertainty. Completion accuracy and NLL concern hidden cells. The NLL in Table 1 and Tables 3–5 is calibrated marginal log loss. For six-round rows, each token is scored at decision time, after earlier predictions may have entered the evidence. Appendix X separately reports first-pass marginal NLL using only original observations and conditional joint NLL for strip components, their mixture and ordered scorers, including the neural comparison in Table 38. The fitted local control and CNN-S are reported as three-training-seed means with standard deviations for accuracy. Other main-table rows are individual fitted models. The standard deviations do not quantify uncertainty over test examples. Paired analyses of NLCD must account for geographic grouping rather than treating nearby or overlapping windows as independent regions. The main comparisons use absolute means and do not claim geographic-population significance from a tile bootstrap.

Baseline definitions. R1/R2/R3 are neighbourhood radii; K4 denotes context order. Local mode is the evaluation harness's 5×5 vote, scored on the same examples. It is not the simpler eight-neighbour implementation used in an earlier check. The local control is a fitted additive categorical predictor. RADD, MAGE and MaskGIT capacity varies with the dataset; the associated state sizes are given in Tables 3–5. RandAR's additional results remain in the separate-cohort comparison, not mixed into the new main table.

Model fitting and checkpoints. Count-fitted joints use zero-initialized L-BFGS on the average regularized likelihood. The four joint fits and the calibration of their conditional pool are distinct operations. CNN-S and the local control have three fitted seeds in the main tables. Earlier checkpoint-selection audits and development reselection determine the retained reference checkpoints; their results are not a reason to select new checkpoints using test outcomes. Main-table mean/standard-deviation conventions supersede the earlier single-checkpoint and half-range reporting for those rows.

Storage and timing. Predictor state, serialized file size, temporary memory, maintained statistics and retained training data are separate quantities. Tables preserve the state convention of their own experiment. The 49 kB pairwise maintenance measurement uses 64-bit integers, whereas a 24.6 kB maintained layout in the context study uses 32-bit integers. Reduced precision requires verifying that all counts fit in range; an old timing is not re-labelled as a measurement of another layout. Access to removed records or their cached contributions is needed to subtract them. Calibration records and SISA checkpoints are additional retained information where specified.

C.1 Implementation details

The settings below describe the default $V = 16$ counted constructions. The $V = 64$ and small-data studies retain their separately stated settings and training budgets.

Representation and codebooks. Image grids use DINOv2 ViT-B/14, loaded as `dinov2_vitb14` from `facebookresearch/dinov2` through `torch.hub`. We use `x_norm_patchtokens` on a 16×16 grid. CIFAR images are resized from 32 to 224 pixels with bicubic interpolation and

ImageNet mean/std normalization. The recorded ImageNet transform is `Resize(224)` with its default interpolation and the same normalization; this extraction path was not re-verified end to end. Codebooks use `MiniBatchKMeans` with 16 clusters, `random_state=42`, `n_init=3`, batch size 1,024 and `max_iter=300`. They are fitted to patch features from the first 5,000 CIFAR training images and up to 10,000 ImageNet training images.

Positive priors. Let $C_i(u) = \sum_{x \in \mathcal{D}} \mathbf{1}[x_i = u]$. The image prior is $\pi_i(u) = (C_i(u)+1)/(N+V)$. NLCD shares the pooled prior $\pi(u) = (\sum_{i \in \Omega} C_i(u) + 1)/(N|\Omega| + V)$ across positions. Both are strictly positive; no probability floor or log clamp is used. The pairwise token marginal in Eq. 2 also uses one pseudocount per category. In the strip-pool implementation, the prior term is subtracted: $a_0 = -w_5$.

Table 2: Implementation settings for the counted predictors. Strip-joint fitting and strip-pool calibration are separate operations.

Component	Setting
Strip-joint fit	Float64, zero initialization; L-BFGS learning rate 1.0, `max_iter=400`, `tolerance_grad=1e-9`, `tolerance_change=1e-12`, history size 20 and strong-Wolfe line search. Regularization $\rho = 10^{-3}$ on pair and unary parameters.
Strip-pool fit	Up to 512 training tiles, NumPy seed 3; full-batch masked cross-entropy, Adam learning rate 0.05, 300 steps, float64. Initial expert weights $a_1 = \cdots = a_4 = 1$ and prior coefficient $a_0 = -3$. Small-data fits remain restricted to the stated subset.
Calibration masks	A central hole with probability 0.5; otherwise a hidden fraction from $\{0.2, 0.5, 0.75, 0.9\}$. Masks use the calibration-sampling generator.
Context hashing	2^{20} buckets per context level, four levels and radius three. Keys use a 64-bit FNV-1a-style hash over observed-neighbour (offset index, token) pairs in fixed distance order, reduced modulo 2^{20}. Offset basis 14695981039346656037; prime 1099511628211.
Context estimator	Absolute discount 0.75, prediction count threshold 25, interpolation weight $\lambda = 0.3$.
Contribution masks	Two per record, seeded by the 64-bit BLAKE2b digest of that record's bytes. Each uses the central-hole/hidden-fraction rule above.
CIFAR dictionary	Retain keys with count at least ten on the 1,500 CIFAR test-partition development tiles also used for decoder/temperature selection. These reference tiles are disjoint from task-training records.
Completion selection	One pass versus six rounds by development hidden-cell accuracy. Post-hoc temperature minimizes development NLL over $\{0.5, 0.6, \ldots, 3.0\}$ without changing the completed trajectory.
Ordered likelihood	Temperature minimizes ordered NLL on 300 development tiles over $\{0.50, 0.55, \ldots, 2.50\}$, separately per model, seed and mask (Appendix X).
Hardware	NVIDIA A100 80GB PCIe GPU; Intel Xeon Silver 4210R CPU.

GPU idle status was not recorded for the original strip deletion timings. The separately reported timing-repeat experiment used an idle GPU.

D COMPLETE EQUAL-DATA PREDICTION RESULTS

Tables 3–5 retain all three mask protocols, the dataset-specific prediction state and training-seed variability. Their central-hole values are summarized in Table 1. The fixed dictionary is a separate low-memory variant, not a replacement for the full default context bank.

Table 3: CIFAR-10 feature-token completion. Hidden-token accuracy (%) and calibrated NLL (nats) on the same 3,000 test tiles. The 16×16 grids have $V = 16$; each mask hides 25% of cells. Counted and CNN-S models use 50,000 task-training tiles.

		Central hole		Off-centre hole		Scattered 25%	
Predictor	State	Acc. ↑	NLL ↓	Acc. ↑	NLL ↓	Acc. ↑	NLL ↓
Local mode (5×5)	—	72.03	0.967	69.59	1.057	79.55	0.600
Pairwise R1	24.6 kB	84.54	0.587	73.64	0.921	82.25	0.574
Fitted local	25.2 kB	84.84 (0.05)	0.572	73.82 (0.09)	0.913	82.45 (0.01)	0.557
Strips	73.7 kB	85.20	0.474	76.28	0.700	83.85	0.482
Strips + K4 contexts	5.4 MB	85.79	0.458	76.47	0.730	83.86	0.470
Strips + fixed dictionary	0.68 MB	85.70	0.463	76.37	0.734	83.59	0.479
CNN-S	30.6 kB	86.42 (0.05)	0.410	76.17 (0.06)	0.700	84.03 (0.07)	0.442
MaskGIT-tiny	105.7 kB	85.90	0.422	77.36	0.604	85.04	0.389
RADD	1.2 MB	86.28	0.400	77.77	0.589	85.96	0.362
MAGE	2.5 MB	86.99	0.409	79.27	0.597	86.22	0.375
MaskGIT-medium	8.7 MB	87.35	0.353	80.17	0.516	86.55	0.341

Parentheses are accuracy standard deviations across three training seeds for CNN-S and the fitted control. Other rows are individual fits. Each row uses its development-selected decoder and temperature. State is the numerical prediction payload at 4 bytes per stored parameter, count or table entry, excluding file overhead, the shared encoder/codebook, working memory and maintained statistics. The full K4 bank is the default; the fixed frequency dictionary is a separate low-memory variant.

Table 4: ImageNet feature-token completion. Hidden-token accuracy (%) and calibrated NLL on 3,000 common 16×16 test grids with $V = 16$. Counted and CNN-S models use the same 200,000-tile training budget. Each mask hides 25% of cells.

		Central hole		Off-centre hole		Scattered 25%	
Predictor	State	Acc. ↑	NLL ↓	Acc. ↑	NLL ↓	Acc. ↑	NLL ↓
Local mode (5×5)	—	63.56	1.273	60.72	1.327	73.18	0.789
Pairwise R1	24.6 kB	72.70	1.052	64.25	1.292	75.89	0.798
Fitted local	25.2 kB	72.98 (0.03)	1.038	64.09 (0.08)	1.289	75.66 (0.04)	0.799
Strips	73.7 kB	73.41	0.889	65.77	1.119	76.58	0.747
Strips + K4 contexts	24.6 MB	74.35	0.887	66.24	1.097	76.67	0.733
CNN-S	30.6 kB	74.97 (0.02)	0.761	66.66 (0.10)	0.986	76.97 (0.06)	0.640
MaskGIT-tiny	105.7 kB	74.14	0.771	67.38	0.912	77.61	0.598
RADD	1.2 MB	74.26	0.799	68.48	0.874	78.41	0.573
MAGE	2.5 MB	75.44	0.773	68.82	0.914	78.44	0.596
MaskGIT-medium	8.7 MB	76.30	0.683	70.26	0.805	79.00	0.551

Parentheses denote accuracy standard deviations over three training seeds. Decoder, temperature and state conventions follow Table 3. The context bank is larger than on CIFAR; its full prediction payload is included.

Table 5: NLCD land-cover completion. Hidden-cell accuracy (%) and calibrated NLL on 3,000 common 32×32 windows with 16 category labels. Counted and CNN-S models use 200,000 training windows. Each mask hides 25% of cells.

		Central hole		Off-centre hole		Scattered 25%	
Predictor	State	Acc. ↑	NLL ↓	Acc. ↑	NLL ↓	Acc. ↑	NLL ↓
Local mode (5×5)	—	73.83	1.088	73.71	1.092	81.22	0.555
Pairwise R3	49.2 kB	73.51	1.042	73.51	1.055	81.45	0.619
Fitted local	26.2 kB	73.13 (0.00)	1.077	73.17 (0.01)	1.077	82.61 (0.00)	0.579
Strips	8.5 kB	74.30	0.957	74.41	0.953	84.32	0.509
Strips + K4 contexts	55.8 MB	74.74	0.926	74.75	0.925	84.37	0.498
CNN-S	30.6 kB	74.33 (0.14)	0.868	74.22 (0.11)	0.873	84.13 (0.06)	0.453
MaskGIT-tiny	204.0 kB	74.56	0.810	74.45	0.815	84.00	0.444
RADD	9.5 MB	75.00	0.704	75.09	0.705	84.73	0.421
MAGE	18.0 MB	75.28	0.731	75.19	0.735	84.77	0.446
MaskGIT-medium	9.5 MB	75.32	0.693	75.52	0.687	84.84	0.418

Parentheses denote accuracy standard deviations over three training seeds; 0.00 is rounding. Strip-only prediction shares unaries. The context bank is a separate, much larger state. RADD and MAGE have larger capacities than their image-token counterparts, as the state column shows. State follows the numerical-payload convention of Table 3. The table reports means, not geographic-population significance.

E PAIRED COMPARISONS ON THE MAIN EVALUATION SETS

Tables 6–8 report the supplied paired accuracy comparisons for the context-extended predictor. For CNN-S and the fitted local control, the reference is the per-tile mean of the three training seeds, matching the main-table aggregation. Other references use their individual fitted models. Resampling is over tiles, conditional on these fits; it does not resample the training runs. On NLCD they do not account for all geographic dependence and are not used to claim significance over a population of independent regions. Last-digit differences from rounded main-table means can arise from rounding. The separate strip-only intervals in Appendix V use earlier training budgets and a single CNN checkpoint; those intervals are not attached to the current equal-data rows.

Table 6: CIFAR: K4 contexts on strips minus each reference. Paired accuracy differences in percentage points, with 95% tile-bootstrap intervals. Seeded references use the per-tile mean of three fits.

Reference	Central	Off-centre	Scattered
Local mode	$+13.76$ $[+13.05, +14.48]$	$+6.89$ $[+6.45, +7.33]$	$+4.31$ $[+4.13, +4.49]$
Pairwise R1	$+1.25$ $[+1.05, +1.45]$	$+2.83$ $[+2.57, +3.08]$	$+1.61$ $[+1.48, +1.75]$
Fitted local	$+0.95$ $[+0.77, +1.11]$	$+2.65$ $[+2.42, +2.88]$	$+1.41$ $[+1.29, +1.53]$
Strips	$+0.58$ $[+0.45, +0.72]$	$+0.19$ $[+0.03, +0.35]$	$+0.01$ $[-0.04, +0.07]$
CNN-S	-0.63 $[-0.78, -0.48]$	$+0.30$ $[+0.09, +0.50]$	-0.17 $[-0.28, -0.05]$
MaskGIT-tiny	-0.12 $[-0.29, +0.05]$	-0.89 $[-1.14, -0.64]$	-1.18 $[-1.32, -1.04]$
RADD	-0.49 $[-0.66, -0.33]$	-1.29 $[-1.53, -1.06]$	-2.10 $[-2.23, -1.96]$
MAGE	-1.20 $[-1.37, -1.04]$	-2.80 $[-3.07, -2.54]$	-2.36 $[-2.51, -2.22]$
MaskGIT-medium	-1.56 $[-1.74, -1.38]$	-3.70 $[-3.98, -3.42]$	-2.68 $[-2.83, -2.54]$

Table 7: ImageNet: K4 contexts on strips minus each reference. Paired accuracy differences in percentage points, with 95% tile-bootstrap intervals. Seeded references use the per-tile mean of three fits.

Reference	Central	Off-centre	Scattered
Local mode	$+10.79$ $[+10.21, +11.37]$	$+5.52$ $[+5.16, +5.88]$	$+3.49$ $[+3.30, +3.68]$
Pairwise R1	$+1.65$ $[+1.45, +1.85]$	$+1.99$ $[+1.73, +2.24]$	$+0.77$ $[+0.64, +0.91]$
Fitted local	$+1.37$ $[+1.18, +1.57]$	$+2.14$ $[+1.91, +2.37]$	$+1.01$ $[+0.87, +1.15]$
Strips	$+0.94$ $[+0.76, +1.12]$	$+0.46$ $[+0.31, +0.62]$	$+0.09$ $[+0.02, +0.16]$
CNN-S	-0.62 $[-0.81, -0.44]$	-0.43 $[-0.64, -0.21]$	-0.31 $[-0.43, -0.18]$
MaskGIT-tiny	$+0.21$ $[-0.04, +0.46]$	-1.14 $[-1.41, -0.88]$	-0.94 $[-1.08, -0.80]$
RADD	$+0.09$ $[-0.13, +0.30]$	-2.24 $[-2.50, -1.98]$	-1.74 $[-1.88, -1.60]$
MAGE	-1.09 $[-1.31, -0.88]$	-2.58 $[-2.86, -2.30]$	-1.77 $[-1.92, -1.62]$
MaskGIT-medium	-1.95 $[-2.18, -1.73]$	-4.03 $[-4.30, -3.75]$	-2.33 $[-2.48, -2.19]$

Table 8: NLCD: K4 contexts on strips minus each reference. Paired accuracy differences in percentage points, with 95% tile-bootstrap intervals. Seeded references use the per-tile mean of three fits.

Reference	Central	Off-centre	Scattered
Local mode	$+0.91$ $[+0.73, +1.08]$	$+1.04$ $[+0.87, +1.21]$	$+3.15$ $[+3.03, +3.26]$
Pairwise R3	$+1.23$ $[+1.05, +1.42]$	$+1.24$ $[+1.07, +1.42]$	$+2.92$ $[+2.82, +3.03]$
Strips	$+0.44$ $[+0.37, +0.52]$	$+0.34$ $[+0.27, +0.41]$	$+0.05$ $[+0.02, +0.07]$
CNN-S	$+0.41$ $[+0.26, +0.55]$	$+0.53$ $[+0.39, +0.67]$	$+0.24$ $[+0.20, +0.28]$
MaskGIT-tiny	$+0.18$ $[+0.02, +0.34]$	$+0.30$ $[+0.14, +0.46]$	$+0.37$ $[+0.31, +0.42]$
RADD	-0.26 $[-0.42, -0.09]$	-0.34 $[-0.50, -0.18]$	-0.36 $[-0.41, -0.32]$
MAGE	-0.54 $[-0.69, -0.40]$	-0.44 $[-0.58, -0.29]$	-0.40 $[-0.44, -0.36]$
MaskGIT-medium	-0.58 $[-0.76, -0.40]$	-0.77 $[-0.93, -0.60]$	-0.48 $[-0.52, -0.43]$

These tile-bootstrap intervals condition on the fitted models and do not account for geographic clustering. They are not used as evidence of significance over independent geographic regions.

F REBUILDING THE REVISED STRIP AND CONTEXT PREDICTORS

The CIFAR deletion experiment removes 4,500 of 45,000 training records and evaluates the revised predictors on the same 2,000-tile central-hole cohort. Tables 9 and 10 report prediction quality, measured update time and state.

Rebuild procedure. We subtract the removed records' strip statistics and, for the context extension, their contributions under the content-seeded masks. We refit the four strip joints from zero, then refit all five pool coefficients using the settings in Appendix C.1. Calibration draws 512 tiles without replacement from the revised training collection using a NumPy generator with seed 3; the same generator supplies the masks. The fresh build uses the same revised collection and sampling rule, so it receives the same calibration inputs and no deleted record can be selected. We then reconstruct the pruned context view at threshold 25. Decoder and post-hoc temperature are re-selected on fixed reference tiles containing no task-training records.

Timing and retained information. The 29.3 s strip measurement includes statistic updating and checks, four joint fits, pool-coefficient fitting and its conditioning passes on the 512 calibration tiles. The 33.8 s context measurement also includes context subtraction and reconstruction of the prediction view. Decoder and temperature re-selection are evaluation-side operations excluded from these times; these are not end-to-end times including model selection. The approximately 24 s four-joint refit is incurred regardless of edit size; context subtraction takes 5.5 s for this event. The 0.1 MB and 121.6 MB inventories contain maintained statistics, not retained training or reference data. Pool recalibration reads a new 512-tile sample from the revised collection, so the full procedure requires access to retained records and is not statistics-only.

Agreement checks. Integer strip and context statistics match their fresh-build counterparts exactly. All four strip parameter vectors are identical ($\max|\Delta\theta| = 0$). Context-predictor log-probabilities were compared on 200 cohort tiles and agree within 10^{-6}. The reported fresh-match status refers to these checks; the log-probability check is numerical agreement on the evaluated tiles, not an exhaustive bitwise comparison of every possible query. Appendix A gives the deterministic-reconstruction argument.

Table 9: CIFAR after deleting 4,500 of 45,000 training records: absolute accuracy on the same 2,000-tile cohort, update or fresh-fit time, and maintained state. Every row uses its development-selected decoder. Prediction payload is a separate quantity.

Procedure	Acc. (%)	Update / fit (s)	Fresh match	Maintained
Counted, fixed calibration	84.18	0.118	Yes	49 kB[c]
Counted, refitted calibration	84.34	1.01	Yes	49 kB[a,c]
Ridge	83.02	0.87	Yes	1.3 MB
Strips, refit[d]	85.14	29.3	Yes	0.1 MB
Strips + contexts, refit[d]	85.60	33.8	Yes	121.6 MB
Fitted local, fresh fit	83.94	9.49	Reference	11.5 MB
CNN-S, fresh fit	85.80	67.91	Reference	11.5 MB
SISA CNN-S (4 shards, 4 slices)	85.57	69.2	Yes	11.5 MB[b]
RADD-small, fresh fit	86.02	282	Reference	—
MaskGIT-tiny, fresh fit	85.69	412.65	Reference	11.5 MB
MaskGIT-small, fresh fit	86.45	496	Reference	—
RandAR-small, fresh fit	85.60	593	Reference	—
MaskGIT-medium, fresh fit	86.77	919	Reference	—
MaskGIT-large, fresh fit	87.65	1,902	Reference	—
MAGE-small, fresh fit	86.71	2,003	Reference	—
CNN-S, warm start	85.94	6.85	No	11.5 MB
MaskGIT-tiny, warm start	85.61	108.12	No	11.5 MB

Reference denotes the declared fresh-build procedure. [a]Calibration inputs are retained separately. [b]Sliced SISA also retains model/optimizer checkpoints. [c]Pairwise statistics use 64-bit integers in this measurement. [d]Strip times include joint fitting and pool recalibration but exclude decoder and temperature selection; maintained-state values exclude retained training/reference data, and pool recalibration reads 512 retained training records. Fresh-match checks for the strip rows compare revised and fresh-build statistics and strip parameters, with predictor log-probabilities agreeing within 10^{-6} on the checked cohort. The context row uses the full bank, not the low-memory dictionary. Dashes denote unreported maintained state. Appendix F gives the complete rebuild, state, and timing definitions.

Table 10: The stronger models on the common CIFAR deletion cohort. Accuracy and calibrated NLL are measured after deletion; both updated models match the declared fresh build. Build and update times include the reported operations of their respective procedures.

Quantity	Count-fitted strips	Strips + contexts (K4)
Accuracy (%)	85.14	85.60
Calibrated NLL	0.481	0.465
Revision time (s)	29.3	33.8
Fresh-build time (s)	69.4	69.4
Prediction state	73.7 kB	4,434.5 kB
Maintained statistics	0.1 MB	121.6 MB

Revision times exclude decoder and temperature selection. Maintained-statistic sizes exclude retained training and reference data; pool recalibration reads 512 records from the revised collection. These are not total retained-data footprints.

G CONTEXT STORAGE AND REVISION

Table 11: Contexts on the pairwise base. Revisable contexts on the CIFAR deletion cohort. Dictionary selection and pruning change prediction state and maintained statistics differently. These are count-family comparisons, all with six confidence rounds.

Predictor	Acc. (%)	Cal. NLL	Prediction	Maintenance
Pairwise R1	84.23	.600	24.6 kB	24.6 kB[a]
Context K4, threshold 25	85.34	.515	4.4 MB	121 MB
Frozen frequency dictionary	85.31	.516	3.9 MB	7.4 MB
Frozen gain dictionary	85.25	.527	1.9 MB	3.7 MB

[a]This experiment stores the base statistics with 32-bit integers; Table 9 retains the 64-bit inventory of its measured procedure (49 kB). The counts are the same; the storage layouts differ. The 7.9 s context-update measurement uses a larger K8 maintenance structure and is not assigned to these smaller dictionaries.

Here the fixed dictionaries use the pairwise base and are selected on independent reference data, either by frequency or held-out predictive gain. The smaller dictionary on the strip base is a separate configuration (Appendix P). Target records subsequently contribute counts to that declared dictionary. The gain-selected dictionary shown here is smaller and slightly less accurate than the frequency-selected one; their budgets differ. With pruning instead of a fixed dictionary, unpruned counts must remain available because revisions can change which contexts cross the prediction threshold.

H BROADER NEURAL AND STATISTICAL COMPARISONS

Tables 12 and 13 report the broader comparison on separate common 2,000-tile sets, with macro-F1, state and decoder details. These sets differ from the 3,000-tile main evaluation. State excludes the encoder and codebook. A dash denotes an unreported measurement.

Table 12: Token completion on 2,000 evaluation tiles per dataset and mask. Each model uses its development-selected decoder. Different architectures are trained on the same token representation; these are not published RGB-generation scores.

Method	Dec.	Acc. (%) ↑	Macro-F1 ↑	Cal. NLL ↓	State kB
(a) CIFAR-10, DINOv2 tokens; central 25% missing					
Local mode	I	72.36	0.566	0.957	0.0
Counted R1	I	84.72	0.668	0.583	24.6
Context counts (R3, K4)	I	85.60	0.681	0.509	2,020.3
Fitted local R1	I	84.95	0.660	0.571	25.2
CNN-S	S	86.57	0.701	0.412	30.6
MaskGIT-tiny	S	86.03	0.680	0.421	105.7
MaskGIT-medium	S	87.52	0.722	0.352	8,731.7
RADD-small	S	86.42	0.696	0.402	1,208.9
RandAR-small	I	86.31	0.688	–	1,330.2
MAGE-small	S	87.14	0.714	0.408	2,535.0
(b) ImageNet, DINOv2 tokens; central 25% missing					
Local mode	I	62.26	0.501	1.314	0.0
Counted R1	I	71.40	0.557	1.106	24.6
Context counts (R3, K4)	I	71.93	0.568	1.046	2,217.7
Fitted local R1	I	71.68	0.553	1.088	25.2
CNN-S	S	73.76	0.578	0.797	30.6
MaskGIT-tiny	S	73.04	0.568	0.805	105.7
MaskGIT-medium	S	75.14	0.599	0.711	8,731.7
RADD-small	S	73.73	0.583	0.778	1,208.9
RandAR-small	S	73.77	0.586	–	1,330.2
MAGE-small	S	74.29	0.595	0.806	2,535.0

S: single pass; I: six rounds. Calibrated NLL is in nats; state is numerical prediction payload in decimal kB. RandAR calibration was not measured. The ImageNet CNN-S row reports the complete epoch-30 evaluation.

Table 13: Token completion: NLCD under two observation patterns.

Method	Dec.	Acc. (%) ↑	Macro-F1 ↑	Cal. NLL ↓	State kB
(a) NLCD land cover; central 25% missing					
Local mode	I	73.14	0.513	1.111	0.0
Counted R1	I	68.46	0.481	1.187	8.2
Counted R2	I	72.59	0.506	1.109	24.7
Fitted local R2	I	72.50	0.507	1.090	26.2
Counted R3	I	73.04	0.507	1.082	49.3
CNN-S	I	73.37	0.495	0.900	30.6
MaskGIT-tiny	S	73.69	0.522	0.751	204.0
MaskGIT-medium	S	74.48	0.527	0.711	9,518.1
RADD-medium	S	74.16	0.523	0.723	9,518.1
MAGE-medium	S	74.47	0.528	0.749	17,954.9
(b) NLCD land cover; random 50% missing					
Local mode	S	80.00	0.653	0.615	0.0
Counted R1	I	81.56	0.676	0.664	8.2
Counted R2	S	81.34	0.677	0.646	24.7
Fitted local R2	S	81.35	0.678	0.635	26.2
Counted R3	S	81.12	0.674	0.648	49.3
CNN-S	S	82.39	0.684	0.510	30.6
MaskGIT-tiny	S	82.36	0.686	0.497	204.0
MaskGIT-medium	S	82.98	0.697	0.474	9,518.1
RADD-medium	S	82.86	0.695	0.478	9,518.1
MAGE-medium	S	82.93	0.696	0.499	17,954.9

Point estimates are shown; some neural rows use a single checkpoint. Uncertainty conventions are described in Appendix C. Static NLCD has no position prior. RandAR was not evaluated in this NLCD common-set export. The two NLCD protocols hide different fractions, so their absolute accuracies are not an equal-mask-size comparison.

I STRIP AND CONTEXT DEVELOPMENT ABLATIONS

Table 14: Conditioning and aggregation play different roles. CIFAR development accuracy (%) / calibrated NLL, on the same 1,500 tiles for all rows. This development ablation is separate from the CIFAR test results in Table 3.

Predictor	Central	Off-centre	Scattered 25%
Pairwise R1	84.89 / .570	74.32 / .896	82.59 / .562
Four strips, unweighted pool	83.62 / .570	74.45 / .824	80.62 / .664
Four strips, calibrated pool	**85.54 / .465**	**76.87 / .677**	**84.17 / .473**

Table 15: Context/base ablations on development data. Entries are accuracy (%) / calibrated NLL. Missing NLL cells are left blank rather than copied from another interpolation weight. These results do not report the frozen K4, $\lambda = .3$ confirmation configuration.

Data	Construction	Central	Off-centre	Scattered
CIFAR	K4 / pairwise, $\lambda = .5$	85.99 / .492	75.42 / .836	83.45 / .502
	K6 / strips, $\lambda = .5$	86.29 / .445	76.46 / —	84.05 / —
	K6 / strips, $\lambda = .3$	86.18 / .443	77.03 / .712	84.33 / .461
ImageNet	K4 / pairwise, $\lambda = .5$	71.89 / 1.036	63.33 / —	75.46 / —
	K4 / strips, $\lambda = .5$	72.65 / .961	64.45 / —	75.56 / —

Table 16: Initial strip-component study on CIFAR development data. Entries are accuracy (%) / *raw* masked marginal NLL. The proper H/V mixture updates component weights using the observed-data likelihood. The pooled product is a conditional heuristic. These numbers are not merged with the later shifted-four-expert study.

Construction	Central	Off-centre	Scattered
One-row chains	80.01 / .654	71.14 / .909	78.70 / .686
Two-row H strips	81.53 / .602	74.88 / .771	82.15 / .555
Two-column V strips	79.56 / .656	73.77 / .782	82.28 / .526
Joint mixture H/V	81.67 / .593	74.74 / .764	82.48 / .526
Pooled H/V marginals	84.65 / .584	76.06 / .850	83.08 / .699
Pairwise, six rounds	85.14 / .696	74.44 / 1.161	82.74 / .719
Pairwise, one pass	55.29 / 1.529	50.27 / 1.615	82.74 / .641

The corresponding calibrated six-round pairwise NLLs are .563, .893 and .549. Conditional joint NLL per hidden cell for H, V and their mixture is .403, .401 and .387 on the central hole; those joint values are not comparable to the marginal-NLL columns above.

These sweeps explain the choice of construction. The frozen model uses K4 and $\lambda = .3$; the exploratory K6 and $\lambda = .5$ values remain separate. This development sweep predates the 32×32 context implementation. The complete NLCD context extension is evaluated in Table 5; it is not inserted into the earlier ablation cohort.

J VOCABULARY SIZE

Table 17 evaluates the frozen $V = 64$ construction on test tiles. Strips improve all three masks over pairwise pooling; they exceed the tested CNN on off-centre and scattered masks but trail it on the central hole. Contexts improve the central hole and slightly reduce accuracy on the other two masks. Their 47.8 MB state is separate from the 393 kB strip state. Table 18 retains the development sweep as a separate evaluation.

Table 17: CIFAR $V = 64$: frozen configurations on 3,000 test tiles. Each score is accuracy (%) / calibrated NLL. These rows are distinct from the development vocabulary sweep.

Predictor	State	Central	Off-centre	Scattered
Pairwise R1	197 kB	61.77 / 1.385	56.41 / 1.685	71.71 / 0.991
Strips	393 kB	66.84 / 1.094	60.85 / 1.318	73.68 / 0.866
Strips + K4 contexts	47.8 MB	67.81 / 1.084	60.40 / 1.404	73.55 / 0.862
CNN-S	77 kB	69.24 / 0.987	58.90 / 1.410	72.20 / 0.894
MaskGIT-tiny	131 kB	64.80 / 1.114	60.36 / 1.284	74.34 / 0.755

All predictors receive the same $V = 64$ tokens. The CNN and MaskGIT rows are individual checkpoints. Prediction state excludes the common pretrained encoder and codebook.

Table 18: CIFAR $V = 64$ development comparison on 1,500 tiles. Accuracy (%) / calibrated NLL; the main vocabulary test uses a separate 3,000-tile set.

Predictor	State	Central	Scattered
Pairwise R1	196.6 kB	63.13 / 1.342	71.79 / 0.983
Strips	393.2 kB	67.92 / 1.069	73.75 / 0.860
Strips + contexts	47.8 MB	68.66 / 1.055	73.61 / 0.856
CNN-S	76.9 kB	69.92 / 0.966	72.36 / 0.891
MaskGIT-tiny	130.6 kB	66.09 / 1.075	74.32 / 0.751

K ADDITIONAL CONTROLS

Table 19: Counted-prefix neural extensions on CIFAR development data: two training seeds, same second-stage network and budget. Frozen switching, specialized continuation and all-hidden-position correction are different procedures.

Procedure	Hole acc. (%)	Scattered acc. (%)	Neural calls
CNN alone, one pass	86.47	83.84	1
Frozen counted then CNN	85.66	83.55	3
Counted prefix, specialized CNN	86.46	84.05	1
Fitted prefix, specialized CNN	86.53	84.03	1
CNN prefix, specialized CNN	86.70	84.38	4
Counted prefix, correction CNN	86.56	83.79	1
Fitted prefix, correction CNN	86.56	83.82	1

Hole correction minus CNN alone: $+0.09$ points, 95% paired interval $[-0.08, +0.25]$. Scattered specialized continuation minus CNN alone: $+0.21$ points, $[+0.08, +0.34]$. These development intervals do not prove equivalence or isolate higher-order learning. On a separate 3,000-tile confirmation set, frozen counted-to-CNN achieves 85.51% versus 86.36% for the existing one-pass CNN; this is a different experiment and budget.

Conditional probability and coding. A fixed order defines a normalized joint $Q_\pi(x) = \prod_t q(x_{\pi_t} \mid x_{\pi_{<t}})$. The causal coding experiment counts and calibrates a separate causal predictor. Its range decoder regenerates probabilities only from the decoded prefix, and all 2,000 evaluated tiles round-trip exactly. Stream rate excludes shared model bytes; a self-contained archive must include them. The smallest tested causal CNN improves rate over counted R3 at fewer predictor bytes, so this is evidence of an implemented density model, not a claimed best compressor. It is not a codec evaluation of the new strip-PoE predictor.

Other tested constructions. Position-conditioned pairwise tables failed to improve the global counted predictor on the central hole with regional marginal correction; fitted regional controls showed modest gains. The reference-trained nonlinear count aggregator did not meet its stated criterion for exploiting target statistics. Counted tree mixtures and the tested loopy-MRF procedures improved some probability metrics without a consistent accuracy gain. In the earlier facies experiment,

Table 20: Causal lossless token coding on 2,000 test tiles per dataset. All streams decode exactly. Serialized bytes include model-file overhead. Decode times are amortized per tile at batch 250, from a single measured run.

Data	Codec	Model bytes	Bits/token	Decode ms	Fit s
CIFAR	Counted R3	41,632	.8852	3.35	6
	Fitted local R3	42,944	.8121	3.09	304
	Causal CNN	14,808	.8789	1.29	9
NLCD	Counted R3	25,120	.8945	14.58	70
	Fitted local R3	26,432	.8798	28.91	897
	Causal CNN	14,808	.8508	4.27	39

Fitted-local improvements over counted coding from R1 to R3 are .0230/.0589/.0731 bits/token on CIFAR and .0038/.0111/.0147 on NLCD. This is an empirical comparison between fitted procedures, not the population pooling error itself.

the local pairwise sampler had a classifier score of 0.984 under random-order sampling, compared with 0.522 for the U-Net. This is a separate dataset and predictor from the strip-based NLCD sampling experiment in Appendix T.

L QUALITATIVE EXAMPLES AND SELECTION

Examples in Figure 2 and Figure 6 are illustrative selections from a seeded, outcome-stratified export, not a random estimate of model performance. For each mask, the export contains six windows where the context-extended predictor matches or exceeds the per-tile three-seed CNN-S mean and two where CNN-S is better by at least two percentage points. Eligible windows have at least three classes in the hidden region. Displayed rows were chosen from this export to illustrate improvements and failures. The displayed CNN-S completion uses seed 42, whereas aggregate comparisons use three-seed means.

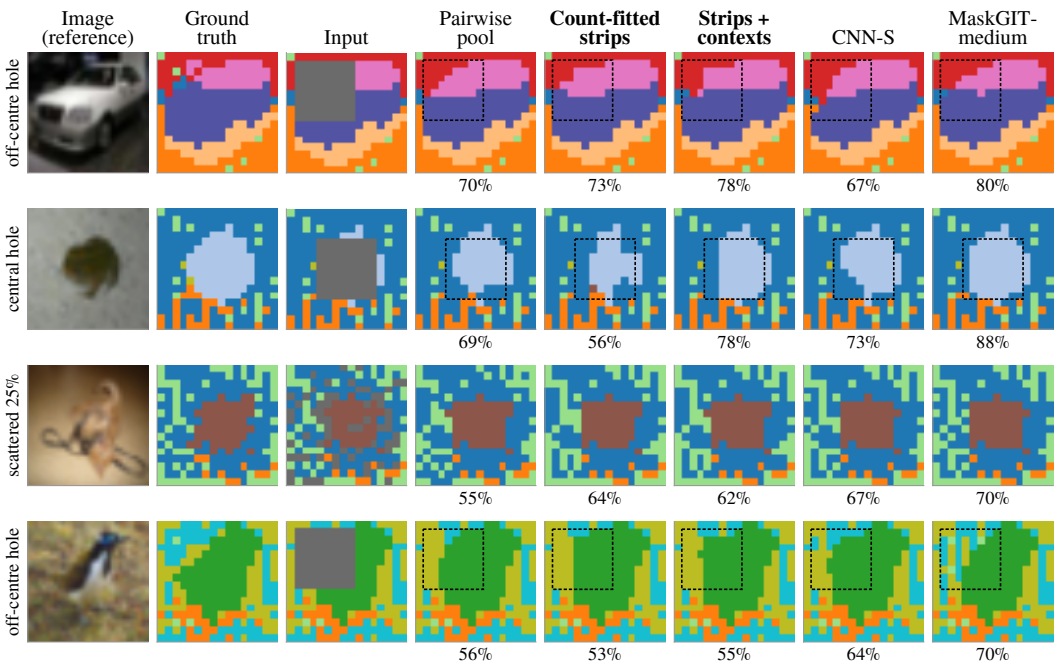


Figure 6: **Feature-token completions on CIFAR.** The RGB image is a reference, not a generated output; predictors complete the precomputed 16×16 grid. Token colors are arbitrary categorical identities. Gray cells are hidden, dashed boxes mark contiguous holes and percentages measure hidden-cell accuracy. The second row illustrates a strip-only failure partly repaired by contexts; the final row favors the neural references. Selection is described above.

M CONTROLLED POOLING ANALYSIS

Table 21: Controlled loopy binary grid, central 2×2 hole, 10^5 independent training states. All values are nats per predicted cell. More context reduces omitted information but can increase the error of unweighted pairwise pooling.

Quantity	4 neighbours	8 neighbours	All observed
Omitted information	.0586	.0209	.0000
Population pooling error	.0037	.0571	.1334
Empirical pooled total	.0622	.0775	.1329
Empirical context-table total	.0586	.0211	.0065

The empirical totals need not equal the first two rows' sum: their finite-sample correction is signed. Exact identity checks have residual at most 1.1×10^{-15}. This test uses genuine observed context and no fitted offset weights. It characterizes pairwise pooling, not the strip construction.

The reported identities separate omitted observations from the error of the unweighted population pool. Empirical estimation adds a signed difference, not a third nonnegative information component. These controlled results characterize pooling; the strip method is evaluated separately on real grids and in its component ablations.

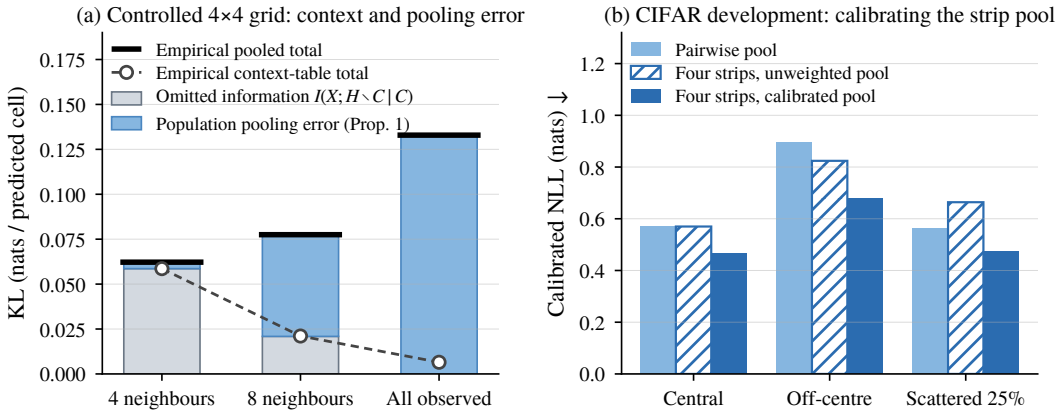


Figure 7: **Two distinct construction diagnostics.** Left: population omitted-information and pooling terms on an enumerable 4×4 grid; black marks are the empirical pooled total, including its signed finite-sample correction. Right: CIFAR development calibration of the four-expert strip pool. These are separate experiments (Tables 21 and 14).

N CHANGED-PAIR MAP UPDATES AND TIMING REPEATS

For map corrections, an edit need not change a whole record. Within one window, only directed pairs with an edited endpoint change:

$$E_S = \{(i, d) : i, i + d \in \Omega,\ i \in S \text{ or } i + d \in S\}, \qquad |E_S| \le 2|\Delta_R||S|. \tag{23}$$

We subtract and add each affected pair once. Overlapping windows retain their original multiplicity; the bound is per window. The same predictor is recovered while avoiding unchanged contributions. Detecting edits, accessing records and reconstructing the predictor remain separate costs.

The map update experiment compares algorithms for the same pairwise statistic on 180,000 windows. The complete times include detecting edits, updating contributions, reconstructing tables and serialization; record reads and calibration refitting are excluded. An edited source cell can appear in several overlapping windows, whose contribution multiplicities are preserved. The advantage is specific to avoiding unaffected pair contributions, not a change in prediction quality.

Table 22: Updating the same R3 statistic on 180,000 overlapping NLCD windows. Times are seconds on one CPU thread and include detection, contribution work, table reconstruction and serialization, but exclude record reads and calibration refitting. All three procedures produce identical integer statistics and reconstructed log tables.

Edit	Changed pairs	Whole windows	Recount
2019–2021 revision	127	361	201
0.1% clustered cells	5	17	174
0.1% scattered cells	40	366	194
10% scattered cells	221	416	138

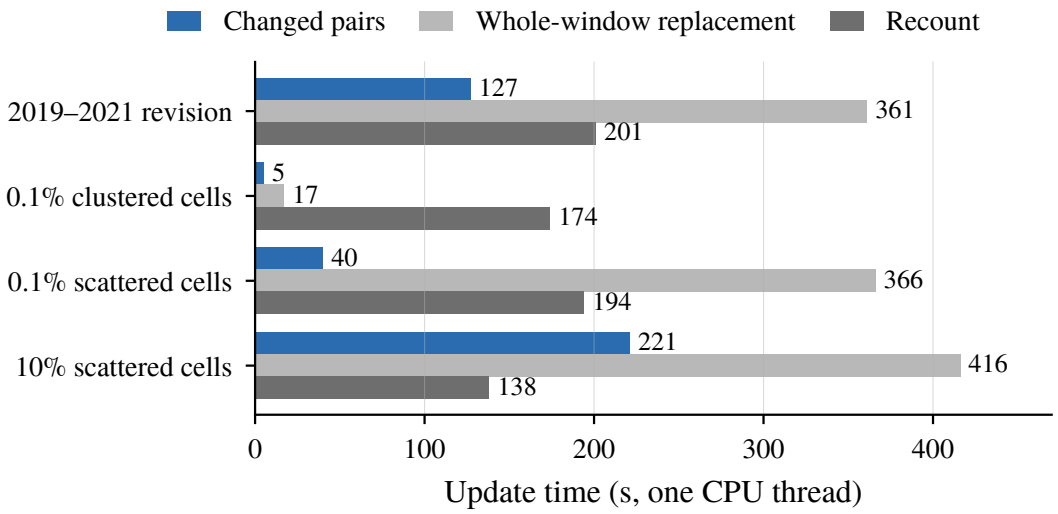


Figure 8: **Updating pairwise map statistics after cell edits.** R3 statistics on 180,000 overlapping NLCD windows, using one CPU thread. Times include detection, contribution updates, table reconstruction and serialization, but exclude record reads and calibration refitting. All procedures produce identical statistics; dense scattered edits can favor recounting. Numerical values are in Table 22.

Table 23: Separate timing-repeat check with the same code and seeds on an idle GPU. Accuracy columns use six-round decoding throughout, unlike the selected single-pass CNN result in Table 9.

Procedure	Original s	Repeat s	Original acc.	Repeat acc.
Pairwise, fixed	0.118	0.106	84.18	84.18
Pairwise, refit	1.013	1.204	84.34	84.34
Ridge, refit	0.869	0.962	83.02	83.02
Control, fresh	9.491	9.056	83.94	83.94
Control, warm	1.565	1.695	84.18	84.18
CNN-S, fresh	67.912	73.262	84.12	84.12
CNN-S, warm	6.852	7.639	83.95	83.95
SISA CNN, 4 shards	71.489	73.183	83.16	83.16

Accuracy repeats exactly at the displayed precision. The timing differences are not all within 8%; this check is not used to replace selected-decoder accuracy or to combine measurements across procedures.

O COMPLETE TASK-TRAINING-SIZE STUDY

Table 24 gives both masks and all reported values behind Figure 3. Accuracy variability is a half-range over three subset draws; calibrated NLL entries are means. The pretrained representation is fixed, and every task-dependent fit stays within the stated budget.

Table 24: CIFAR task-training-size study on 1,500 development tiles. Each entry is accuracy (%), its half-range over three subset draws in parentheses, and calibrated NLL. The pretrained representation is fixed.

Predictor	$N = 100$	$N = 300$	$N = 1{,}000$
Central hole			
Pairwise R1	84.51 (0.07) / 0.605	84.67 (0.02) / 0.588	84.93 (0.10) / 0.573
Fitted local	84.93 (0.25) / 0.597	85.19 (0.01) / 0.566	85.20 (0.04) / 0.562
Strips	84.39 (0.03) / 0.528	85.29 (0.18) / 0.479	85.51 (0.03) / 0.470
Strips + K4 contexts	84.72 (0.15) / 0.518	85.50 (0.08) / 0.482	85.83 (0.08) / 0.468
CNN-S	81.37 (1.89) / 0.661	84.24 (0.49) / 0.511	86.22 (0.15) / 0.428
Scattered 25%			
Pairwise R1	82.46 (0.10) / 0.581	82.51 (0.05) / 0.571	82.63 (0.04) / 0.562
Fitted local	82.27 (0.12) / 0.583	82.65 (0.08) / 0.556	82.73 (0.01) / 0.550
Strips	83.26 (0.05) / 0.515	83.87 (0.03) / 0.486	84.05 (0.03) / 0.478
Strips + K4 contexts	83.26 (0.05) / 0.518	83.52 (0.02) / 0.495	83.73 (0.03) / 0.482
CNN-S	81.88 (0.22) / 0.546	83.03 (0.03) / 0.486	84.00 (0.09) / 0.448

Count and aggregation fits use only the stated N task-training tiles; the fixed pretrained representation is shared. CNN-S uses 12,000 training steps. Half-range is not a standard deviation or a confidence interval. Development-selection and fitting details are in Appendix C.

P CONTEXT-MEMORY TRADE-OFF ON THE STRIP BASE

The CIFAR study fixes $K = 4$ and $\lambda = 0.3$. Among banks within 0.15 percentage points of the full bank's development central-hole accuracy, the declared rule selects the smallest prediction view and the smallest maintained dictionary. It selects threshold 400 and a reference-frozen frequency dictionary with minimum reference count ten. The former prunes only the prediction view; the latter restricts which target-data counts need to be maintained. Their tests remain separate from the full default model and from the pairwise-base deletion study in Appendix G. The fixed dictionary

Table 25: CIFAR context-memory selection on development tiles, using the strip base with $K = 4$ and $\lambda = 0.3$. Scores are accuracy (%) / calibrated NLL. Prediction time is the reported ms per tile; storage categories are separate.

Dictionary/view	Central	Scattered	Deployed	Maintained	ms
Strips	85.54 / 0.465	84.17 / 0.473	0.07 MB	0.1 MB	0.8
$\tau = 400$	86.07 / 0.451	83.93 / 0.470	0.56 MB	126.5 MB	9.9
$\tau = 100$	86.14 / 0.447	84.05 / 0.465	1.58 MB	126.5 MB	9.6
$\tau = 25$ (full)	86.14 / 0.444	84.20 / 0.460	5.36 MB	126.5 MB	10.2
Fixed freq. ≥ 10	86.08 / 0.450	83.94 / 0.469	0.68 MB	1.3 MB	9.7
Fixed freq. ≥ 3	86.17 / 0.447	84.06 / 0.465	2.18 MB	4.1 MB	9.7
Fixed freq. ≥ 1	86.17 / 0.445	84.19 / 0.461	11.65 MB	22.0 MB	9.6
Fixed gain	85.92 / 0.453	84.07 / 0.469	1.14 MB	2.2 MB	9.7

reduces maintained statistics to 1.3 MB and retains 85.70% central-hole accuracy, versus 85.79% with the full bank. Neither reduced bank improves scattered accuracy over strips alone. This comparison measures prediction and storage; it does not measure a complete deletion time for the fixed dictionary.

Table 26: Storage variants on the 3,000 CIFAR confirmation tiles. Scores are accuracy (%) / calibrated NLL. The two reduced banks were selected by a declared development rule, not by these scores.

Predictor	Central	Off-centre	Scattered
Strips	85.20 / 0.474	76.28 / 0.700	83.85 / 0.482
$\tau = 400$	85.66 / 0.464	76.40 / 0.734	83.58 / 0.480
$\tau = 25$ (full)	85.79 / 0.458	76.47 / 0.730	83.86 / 0.470
Fixed freq. ≥ 10	85.70 / 0.463	76.37 / 0.734	83.59 / 0.479

Q CONDITIONING ON INCREASING EVIDENCE

A hidden target is fixed in each of 500 development grids. Nested sets of 1, 2, 4, 8, 32, 128 and all non-target cells are revealed. Every prediction conditions on the actual revealed values, not on other model-generated guesses. Tables 27 and 28 give all measurements underlying Figure 4. Strips

Table 27: CIFAR: one fixed hidden target with nested revealed sets on 500 development grids. NLL is in nats and accuracy in percent. No guessed token is added to the observations. "All" denotes every non-target cell.

Predictor	Metric	1	2	4	8	32	128	All
Marginal	NLL	2.125	2.125	2.125	2.125	2.125	2.125	2.125
	Acc.	36.2	36.2	36.2	36.2	36.2	36.2	36.2
Pairwise	NLL	2.179	2.123	2.050	1.863	1.101	0.445	0.530
	Acc.	37.2	39.0	41.2	46.2	69.6	88.8	90.6
Strips	NLL	1.863	1.641	1.340	0.960	0.456	0.307	0.273
	Acc.	44.6	52.2	61.4	72.4	85.2	90.6	93.2
CNN-S	NLL	2.081	1.778	1.378	0.901	0.448	0.300	0.306
	Acc.	43.6	50.2	59.4	72.4	84.6	91.4	90.2

Table 28: NLCD: one fixed hidden target with nested revealed sets on 500 development grids. NLL is in nats and accuracy in percent. No guessed token is added to the observations. "All" denotes every non-target cell.

Predictor	Metric	1	2	4	8	32	128	All
Marginal	NLL	2.141	2.141	2.141	2.141	2.141	2.141	2.141
	Acc.	31.8	31.8	31.8	31.8	31.8	31.8	31.8
Pairwise	NLL	2.130	2.115	2.080	2.020	1.720	1.132	2.086
	Acc.	32.2	32.6	35.0	37.0	50.4	65.2	72.6
Strips	NLL	2.096	2.038	1.943	1.761	1.256	1.005	0.697
	Acc.	33.6	36.0	40.2	47.2	63.8	69.4	80.2
CNN-S	NLL	2.305	2.233	2.096	1.845	1.177	0.860	0.557
	Acc.	33.6	36.2	41.6	49.4	63.4	71.2	79.4

improve their NLL as observations are added in both datasets. The pairwise pool is not monotone in NLL, even when its accuracy improves. Comparisons with CNN-S depend on evidence and metric: the strips have lower CIFAR NLL with one to four revealed cells, whereas the CNN has lower NLL at eight, 32 and 128 observations. On NLCD the CNN has higher accuracy at four and eight observations and lower NLL from 32 onward.

R PREDICTION BY DISTANCE TO OBSERVATIONS

Distances are measured to the originally observed cells using the Chebyshev metric. They are not recomputed from progressively filled guesses. Tables report accuracy and NLL on development tiles;

the spatial composition of each distance group differs. Thus an increase in accuracy toward a hole's centre does not imply that more distant evidence is intrinsically easier to use. In sparse scattered-mask distance groups, no population claim is made without group counts. The CIFAR strip improvement

Table 29: CIFAR development: accuracy (%) / NLL (nats) by Chebyshev distance to the nearest originally observed cell. The positions in the distance groups differ.

Distance	Pairwise	Strips	Strips + contexts	CNN-S
Central hole				
1	83.7 / 0.59	83.9 / 0.50	84.3 / 0.48	85.2 / 0.44
2	84.7 / 0.57	85.2 / 0.45	86.2 / 0.43	86.9 / 0.38
3	86.6 / 0.56	88.1 / 0.45	88.6 / 0.42	88.7 / 0.39
4	89.5 / 0.46	90.8 / 0.36	91.4 / 0.35	91.3 / 0.31
Off-centre hole				
1	80.1 / 0.72	82.1 / 0.55	82.2 / 0.57	82.4 / 0.54
2	70.7 / 1.01	73.4 / 0.75	73.9 / 0.78	73.0 / 0.78
3	68.3 / 1.06	72.0 / 0.80	72.2 / 0.85	71.7 / 0.82
4	70.0 / 1.08	72.2 / 0.81	72.0 / 0.88	70.9 / 0.86
Scattered 25%				
1	82.6 / 0.56	84.2 / 0.47	84.2 / 0.46	84.2 / 0.44
2	50.0 / 1.92	77.6 / 0.68	79.3 / 0.74	74.1 / 0.61

Table 30: NLCD development: accuracy (%) / NLL (nats) by Chebyshev distance to the nearest originally observed cell.

Distance	Pairwise	Strips	Strips + contexts	CNN-S
Central hole				
1	74.4 / 0.93	76.0 / 0.85	77.1 / 0.79	76.1 / 0.72
2	71.4 / 1.08	71.4 / 1.02	71.8 / 0.97	71.4 / 0.86
3	69.4 / 1.17	69.2 / 1.11	69.3 / 1.06	69.2 / 0.96
4	67.5 / 1.28	67.3 / 1.18	67.2 / 1.17	67.5 / 1.05
5	66.1 / 1.33	66.0 / 1.23	65.9 / 1.23	66.1 / 1.15
6	65.2 / 1.38	65.4 / 1.25	65.3 / 1.26	65.0 / 1.32
7	65.7 / 1.40	65.1 / 1.25	65.2 / 1.26	65.1 / 1.35
8	65.3 / 1.43	65.3 / 1.26	65.4 / 1.26	64.3 / 1.38
Off-centre hole				
1	74.8 / 0.95	76.5 / 0.84	77.5 / 0.79	76.3 / 0.73
2	71.0 / 1.10	71.5 / 1.03	71.8 / 0.98	71.1 / 0.88
3	68.9 / 1.19	69.0 / 1.12	69.1 / 1.07	68.7 / 0.98
4	67.1 / 1.29	67.6 / 1.18	67.6 / 1.17	67.1 / 1.06
5	66.0 / 1.34	66.2 / 1.22	66.1 / 1.22	65.5 / 1.16
6	65.1 / 1.39	65.2 / 1.25	65.2 / 1.26	64.7 / 1.33
7	64.1 / 1.43	64.9 / 1.26	65.0 / 1.27	63.7 / 1.37
8	63.6 / 1.44	65.2 / 1.26	65.1 / 1.27	63.5 / 1.38
Scattered 25%				
1	78.9 / 0.70	81.8 / 0.58	81.8 / 0.56	81.5 / 0.51
2	64.8 / 1.33	64.8 / 1.11	66.2 / 1.09	67.6 / 0.88

is larger in the deeper central-hole groups than at the border. On NLCD, central-hole accuracy gains are concentrated at the border, while NLL improvements persist deeper inside. These comparisons are descriptive because the positions in the distance groups differ.

S PAIRING LARGER, TWO-TOKEN PARTS

The factorized strip transition uses separate token-pair factors across adjacent columns. The extension replaces them by a full $V^2 \times V^2$ transition between two-token column states. This representation preserves the original prediction target, but permits fourth-order interactions among its cells. Three regularization values are evaluated on development data, and $\lambda_T = 10^{-4}$ is selected. Initialization reproduces the factorized model. The shifted wrap strip retains its factorized transition. The gains in this test are modest: on CIFAR the extension reaches 85.34%, 76.55% and 84.14%, compared with 85.20%, 76.28% and 83.85% for factorized strips. NLCD gains are smaller. In development fits, the richer strip payload is 1,122 kB on CIFAR and 1,057 kB on NLCD, versus 74 kB and 8 kB for the respective references. We retain factorized strips as the paper's default. The ten-percent deletion checks report equal statistics and repeated parameter fits agreeing exactly, with component-refit times of 22 s and 42 s. These are not complete revision timings for the context-extended predictor.

Table 31: Pairing two-token parts: confirmation accuracy (%) / calibrated NLL. The unrestricted column-state transition is initialized at the factorized solution; $\lambda_T = 10^{-4}$ is selected on development data.

Dataset	Predictor	Central	Off-centre	Scattered
CIFAR	Strips	85.20 / 0.474	76.28 / 0.700	83.85 / 0.482
CIFAR	Two-token parts	85.34 / 0.459	76.55 / 0.711	84.14 / 0.463
NLCD	Strips	74.30 / 0.957	74.41 / 0.953	84.32 / 0.509
NLCD	Two-token parts	74.36 / 0.942	74.41 / 0.940	84.37 / 0.489

The general transition is pairwise over two-token parts and fourth-order over original tokens. The shifted wrap strip retains its factorized transition.

T CONDITIONAL SAMPLING FROM THE FITTED PREDICTORS

The NLCD pilot uses 100 development windows and ten draws per condition, with 75%, 50% or 25% of cells observed. It compares independent marginal draws, forward-filtering backward-sampling within horizontal or vertical strip joints, a proper orientation mixture selected by observed-evidence likelihood, and sequential random-order draws from the calibrated pool or CNN-S. Joint strip sampling preserves dependencies within strips but not between independent strips. Sequential pooling follows the order-dependent distribution induced by its scorer, conditioning on observations and preceding draws. It is not assumed to be a conditional of an order-independent full-grid joint.

Table 32: Corrected NLCD sampling critic on 100 development windows, ten draws per condition. Each entry is the classifier two-sample score / its permutation reference. True windows and completions use the same sources, grouped for critic training and evaluation.

Sampler	75% observed	50% observed	25% observed
Independent pool draws	0.477 / 0.442	0.778 / 0.425	0.914 / 0.432
Joint horizontal strips	0.698 / 0.428	0.858 / 0.433	0.945 / 0.471
Joint vertical strips	0.716 / 0.422	0.876 / 0.438	0.933 / 0.445
Joint orientation mixture	0.672 / 0.430	0.846 / 0.408	0.931 / 0.461
Sequential pool draws	0.474 / 0.426	0.728 / 0.425	0.844 / 0.430
Sequential CNN-S draws	0.456 / 0.426	0.532 / 0.440	0.692 / 0.432

The nominal chance score is 0.5; the empirical permutation references are reported alongside each score.

The critic uses true windows and their completions from the same source groups. An earlier geographic-half comparison failed its real-versus-real control and is excluded. In the corrected pilot, sequential pooled samples have a score of 0.474 with 75% observed; their scores rise to 0.728 and 0.844 as observations decrease. The corresponding CNN scores are 0.456, 0.532 and 0.692. Independent strip components are more readily distinguished. The count-based sampler becomes

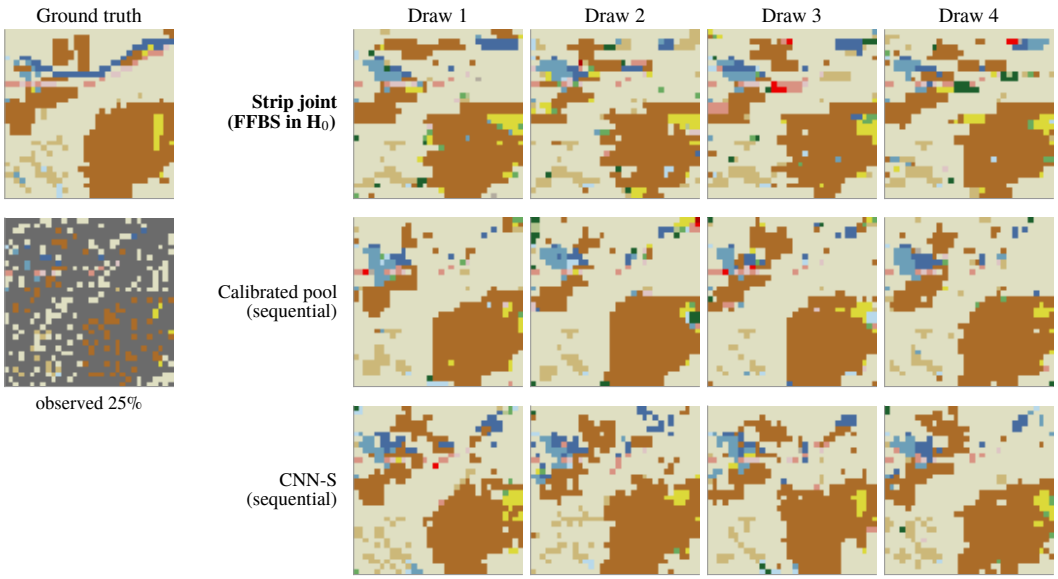


Figure 9: **Conditional NLCD samples with 25% observed and 75% hidden.** The H_0 component uses forward-filtering backward-sampling, jointly within each horizontal strip and independently across strips. The calibrated pool and CNN-S use random-order sequential draws conditioned on observations and preceding draws. This sheet illustrates individual realizations; aggregate critic measurements are reported in Table 32.

more distinguishable as observations decrease, and the sequential CNN has lower critic scores at 50% and 25% observation. The pilot is small, and the empirical permutation references depart from 0.5, so the scores characterize this critic rather than establish distributional equivalence.

U Encoder observation-support audit

The image task masks a precomputed token grid. It does not claim that the supplied tokens could be extracted using only the visible RGB pixels. To check the distinction, the pixels under a central hole in 500 CIFAR images are replaced and the images re-encoded. The changed-token fractions extend beyond the one-token border. Together with the 97.3% no-edit re-encoding agreement, these measurements show why visible feature tokens cannot be treated as isolated encodings of visible pixels. All image predictors share the stored grid in the reported completion task. The completion scores are evaluated on the shared stored token grids; the audit reports the behavior of re-encoding separately.

Table 33: Observation-support audit on 500 CIFAR images. Pixels under the nominal central hole are replaced before DINOv2 re-encoding. Entries are percentages of visible tokens that change. Re-encoding without replacement agrees with the stored tokens on 97.3% of cells.

Replacement	Visible changed	One-token ring	Farther
Random noise	20.6%	35.2%	17.2%
Mean grey	18.9%	27.1%	17.0%
Mean colour of the visible region	19.0%	26.5%	17.3%

The no-edit disagreement is reported separately, not subtracted from the replacement rates. These checks distinguish encoded-grid completion from raw-pixel inpainting.

V ADDITIONAL STRIP-ONLY UNCERTAINTY CHECKS

These checks use the single CNN-S seed-42 checkpoint. The ImageNet and NLCD strip/pairwise models in these checks use the earlier 50,000-record count collection. They are retained with their own identities, not presented as confidence intervals for the equal-data, three-seed means in the main tables. The NLCD geographic bootstrap resamples source cells; its widths cannot be transferred to another fit without recomputing predictions.

Table 34: CIFAR strip-only contrasts against the original single-checkpoint reference. These tile-bootstrap intervals use CNN-S seed 42, not the three-seed mean in the main table.

Mask	Reference	Difference [95%]
Central	Pairwise	$+0.67[+0.46, +0.87]$
Central	Contexts on strips	$-0.58[-0.72, -0.45]$
Central	CNN-S (seed 42)	$-1.15[-1.33, -0.98]$
Off-centre	Pairwise	$+2.64[+2.39, +2.89]$
Off-centre	Contexts on strips	$-0.19[-0.35, -0.03]$
Off-centre	CNN-S (seed 42)	$+0.03[-0.19, +0.26]$
Scattered	Pairwise	$+1.60[+1.47, +1.73]$
Scattered	Contexts on strips	$-0.01[-0.07, +0.04]$
Scattered	CNN-S (seed 42)	$-0.10[-0.23, +0.03]$

Table 35: Additional ImageNet strip-only check with 50,000 count-training records and CNN-S seed 42. This earlier comparison is not the equal-data main-table result.

Mask	Reference	Difference [95%]
Central	Pairwise	$+0.33[+0.09, +0.57]$
Central	Contexts on strips	$-1.26[-1.44, -1.08]$
Central	CNN-S (seed 42)	$-2.01[-2.26, -1.76]$
Off-centre	Pairwise	$+1.03[+0.75, +1.31]$
Off-centre	Contexts on strips	$-0.90[-1.08, -0.72]$
Off-centre	CNN-S (seed 42)	$-1.67[-1.95, -1.39]$
Scattered	Pairwise	$+0.85[+0.71, +0.99]$
Scattered	Contexts on strips	$-0.01[-0.08, +0.07]$
Scattered	CNN-S (seed 42)	$-0.46[-0.59, -0.32]$

Table 36: Additional NLCD strip-only check: 50,000 count-training windows and CNN-S seed 42. Geographic-cell and tile bootstrap intervals concern this earlier configuration, not the current 200,000-window strip model.

Mask	Reference	Difference	Tile 95%	Geographic 95%
Central	Pairwise	$+0.86$	$[+0.69, +1.03]$	$[+0.65, +1.08]$
Central	Contexts on strips	-0.39	$[-0.50, -0.27]$	$[-0.53, -0.25]$
Central	CNN-S (seed 42)	$+0.16$	$[-0.01, +0.34]$	$[-0.05, +0.37]$
Off-centre	Pairwise	$+0.97$	$[+0.79, +1.14]$	$[+0.69, +1.24]$
Off-centre	Contexts on strips	-0.32	$[-0.45, -0.19]$	$[-0.48, -0.18]$
Off-centre	CNN-S (seed 42)	$+0.33$	$[+0.14, +0.51]$	$[+0.11, +0.56]$
Scattered	Pairwise	$+2.99$	$[+2.88, +3.11]$	$[+2.43, +3.58]$
Scattered	Contexts on strips	-0.15	$[-0.18, -0.12]$	$[-0.25, -0.08]$
Scattered	CNN-S (seed 42)	$+0.13$	$[+0.08, +0.19]$	$[+0.02, +0.24]$

W PREDICTION QUALITY AND STATE BY MASK

Figures 10–12 visualize Tables 3–5. Each figure uses one mask, rather than averaging over protocols. State is prediction payload: the shared representation, working memory and statistics retained for revision are excluded. Arrows distinguish changes in construction, not a continuously tunable frontier. The local-vote reference has no learned prediction state.

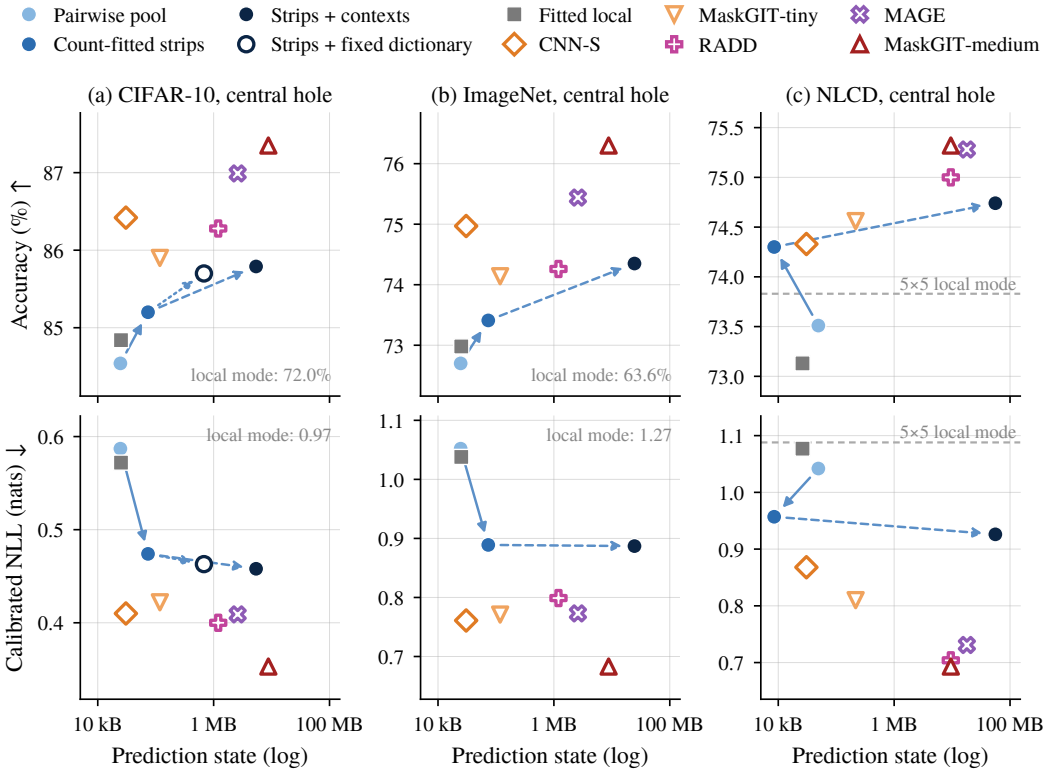


Figure 10: **Prediction quality versus state: central hole.** Top: hidden-token accuracy. Bottom: calibrated marginal NLL. Solid arrows change pairwise pooling to strips; dashed arrows add higher-order contexts; dotted arrows show the separate fixed CIFAR dictionary. Markers retain the strong neural references. All numbers are from the equal-data tables.

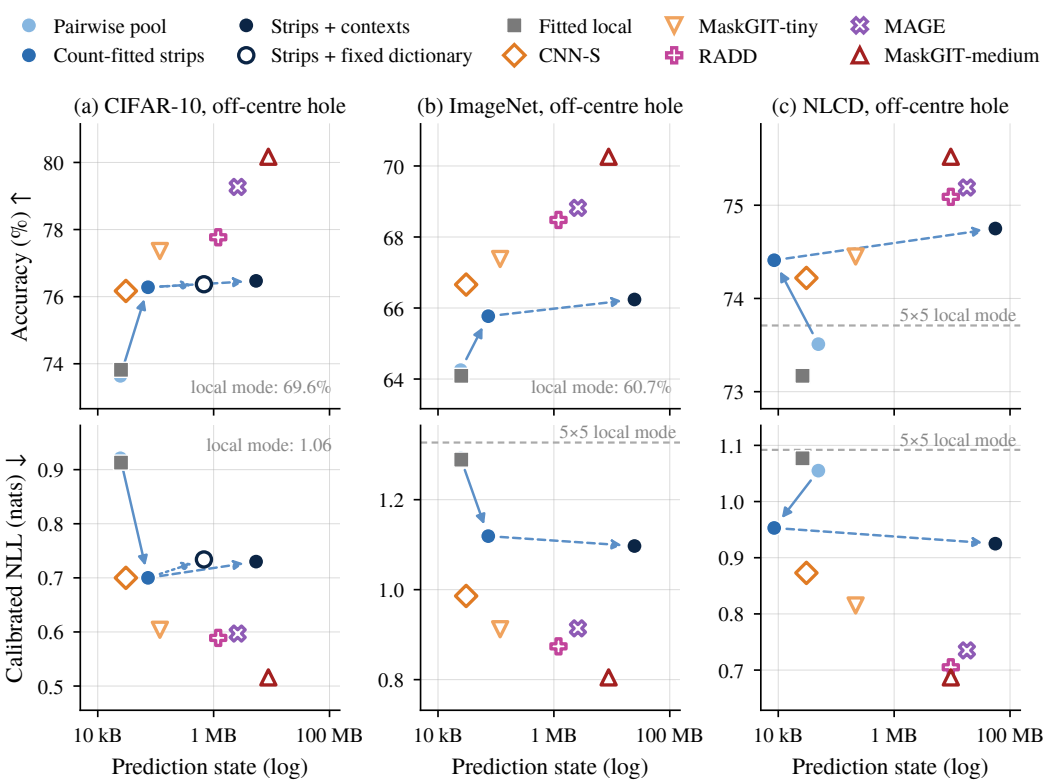


Figure 11: **Prediction quality versus state: off-centre hole.** Top: hidden-token accuracy. Bottom: calibrated marginal NLL. Solid arrows change pairwise pooling to strips; dashed arrows add higher-order contexts; dotted arrows show the separate fixed CIFAR dictionary. Markers retain the strong neural references. All numbers are from the equal-data tables.

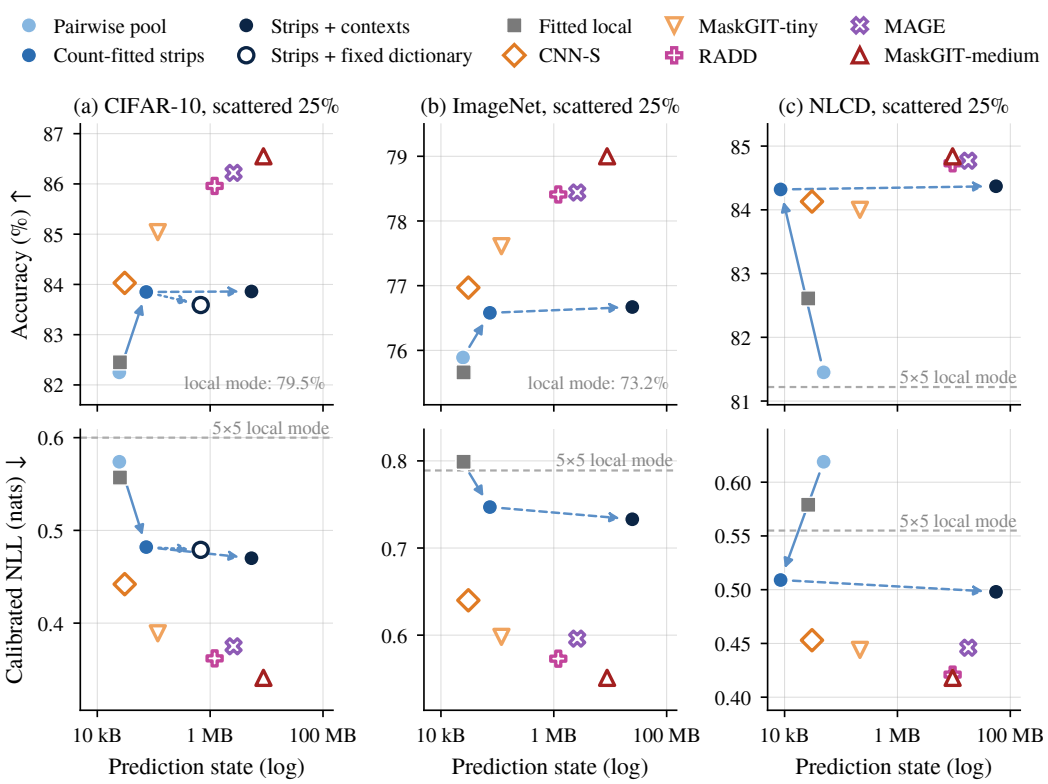


Figure 12: **Prediction quality versus state: scattered 25%.** Top: hidden-token accuracy. Bottom: calibrated marginal NLL. Solid arrows change pairwise pooling to strips; dashed arrows add higher-order contexts; dotted arrows show the separate fixed CIFAR dictionary. Markers retain the strong neural references. All numbers are from the equal-data tables.

X JOINT LIKELIHOOD OF STRIP COMPONENTS AND THE ORDERED POOL

We evaluate the frozen strip components, strip pool and neural scorers on the same 3,000 test tiles per dataset and the same three masks as Tables 3–5. Each mask hides 25% of cells. All losses below are in nats per hidden cell. The strip pool is evaluated without higher-order contexts.

Component and mixture likelihoods. Each component in Eq. 5 is a normalized joint over the grid, factorized across its disjoint strips. Let $Z_m(x_A)$ denote its partition function with the values at positions A clamped. Since $\mathcal{O} \cup \mathcal{U} = \Omega$, its conditional joint probability is

$$q_m(x_\mathcal{U} \mid x_\mathcal{O}) = \frac{Z_m(x_{\mathcal{O}\cup\mathcal{U}})}{Z_m(x_\mathcal{O})}, \qquad \ell_{\text{joint},m} = -\frac{1}{|\mathcal{U}|} \log q_m(x_\mathcal{U} \mid x_\mathcal{O}). \tag{24}$$

The clamped partition functions are computed by the recursion in Appendix B. The proper mixture $q_{\text{mix}}(x) = \sum_{m=1}^{4} \omega_m q_m(x)$ uses a uniform component prior, $\omega_m = 1/4$. After conditioning, its component weights are proportional to $\omega_m q_m(x_\mathcal{O})$; they are not generally uniform. Original observations remain fixed throughout. An alternative prior fitted by expectation–maximization on 512 training tiles changes these losses by at most 0.001 nats per hidden cell; all reported mixture values use the uniform prior. Table 37 also reports the lowest individual-component loss for each dataset and mask. This is a descriptive minimum of four test scores, not a development-selected component.

Ordered likelihood and neural comparison. For a scorer f, let $q_{f,i}^{(\tau)}$ be its token distribution after dividing its logits by a positive temperature τ and normalizing. For the strip pool, the logits are those

in Eq. 7. A fixed raster order π over the hidden cells defines

$$Q^{f}_{\pi,\tau}(x_{\mathcal{U}} \mid x_{\mathcal{O}}) = \prod_{t=1}^{|\mathcal{U}|} q^{(\tau)}_{f,\pi_t}(x_{\pi_t} \mid x_{\mathcal{O}}, x_{\pi_{<t}}) \,. \tag{25}$$

We score this joint by teacher forcing: each factor conditions on the original observations and true preceding hidden tokens, not on model predictions. The reported ordered NLL is $-|\mathcal{U}|^{-1} \log Q^{f}_{\pi,\tau}(x_{\mathcal{U}} \mid x_{\mathcal{O}})$. We apply the same order and observation sets to the strip pool, CNN-S, MaskGIT-tiny and MaskGIT-medium (Chang et al., 2022). These are ordered distributions induced by the evaluated scorers, not likelihoods taken from the original models' published benchmarks. Normalization follows from the sequential factorization; the masked conditionals need not belong to one common order-independent joint.

We report $\tau = 1$ and a temperature $\widehat{\tau}$ selected separately for each model, training seed and mask by minimizing ordered NLL on 300 development tiles over $\{0.50, 0.55, \ldots, 2.50\}$. CNN-S results average the losses of three separately evaluated training seeds; they are not ensemble scores. Development-selected temperatures are retained even when test NLL is slightly higher than at $\tau = 1$.

Marginal scores. The first-pass marginal NLL of scorer f is $-|\mathcal{U}|^{-1} \sum_{i \in \mathcal{U}} \log q^{(1)}_{f,i}(x_i \mid x_{\mathcal{O}})$. It uses only original observations, no intermediate tokens, and temperature one. The completion-table NLL instead uses the development-selected decoder and post-hoc temperature. For the six-round strip rows (ImageNet off-centre and both NLCD holes), it records each token's probability when the token is filled, potentially after other predictions have become evidence. Table 37 separates these strip-model scores. Table 38 gives the complete ordered and first-pass comparison with neural references.

Table 37: Joint and marginal likelihoods of the strip constructions on the common 3,000-tile test sets. All values are nats per hidden cell; lower is better. The strip pool excludes higher-order contexts. The separate neural comparison is in Table 38.

		Conditional joint NLL			Marginal NLL	
Dataset	Mask	Mixture	Best component	Ordered pool $\widehat{\tau}$ ($\tau = 1$)	First pass $\tau = 1$	Selected decoder
CIFAR-10	Central	0.390	0.397 (H_1)	0.338 (0.351)	0.479	0.474
	Off-centre	0.543	0.550 (V_0)	0.480 (0.489)	0.700	0.700
	Scattered	0.521	0.520 (V_0)	0.469 (0.479)	0.489	0.482
ImageNet	Central	0.763	0.774 (V_1)	0.686 (0.711)	0.891	0.889
	Off-centre	0.888	0.902 (V_0)	0.816 (0.823)	1.075	1.119
	Scattered	0.793	0.797 (V_0)	0.737 (0.756)	0.760	0.747
NLCD	Central	0.648	0.655 (V_1)	0.565 (0.578)	0.876	0.957
	Off-centre	0.647	0.654 (H_1)	0.564 (0.577)	0.876	0.953
	Scattered	0.548	0.559 (V_0)	0.491 (0.507)	0.523	0.509

Every mask hides 25% of cells. The mixture uses a uniform prior over its four components and conditions on the original observations. The best-component column is the minimum of four test NLLs, not a selected predictor. Parentheses in the ordered-pool column show $\tau = 1$; preceding values use the development-selected $\widehat{\tau}$. First-pass marginal NLL uses only original observations at $\tau = 1$. Selected-decoder marginal NLL reproduces the strip-only rows in Tables 3–5, including their decoder and temperature settings.

Results. The ordered strip pool improves on the proper component mixture in all nine settings, both at temperature one and with a development-selected temperature (Table 37). Every neural reference has lower ordered joint NLL than the strip pool in each setting, with MaskGIT-medium lowest among the compared scorers (Table 38).

The first-pass comparison has a different ordering in some settings. At temperature one, strips have lower marginal NLL than CNN-S on the two NLCD holes: 0.876 on both masks, versus 0.974 and 0.977. CNN-S has lower first-pass NLL on scattered NLCD observations, and both MaskGIT

references have lower first-pass NLL on all three NLCD masks. These untempered forecasts are distinct from the calibrated, selected-decoder probabilities in the completion tables.

Table 38: Ordered joint likelihood and first-pass marginal NLL for the strip pool and neural scorers on the same 3,000 test tiles per dataset. Every mask hides 25% of cells. Values are nats per hidden cell; lower is better. Ordered entries show development-selected-temperature NLL, with temperature-one NLL in parentheses.

	Ordered joint NLL: $\widehat{\tau}$ ($\tau = 1$)			First-pass marginal NLL: $\tau = 1$		
Predictor	Central	Off-centre	Scattered	Central	Off-centre	Scattered
CIFAR-10						
Strip pool	0.338 (0.351)	0.480 (0.489)	0.469 (0.479)	0.479	0.700	0.489
CNN-S (3 seeds)	0.313 (0.313)	0.437 (0.437)	0.437 (0.438)	0.410	0.707	0.442
MaskGIT-tiny	0.302 (0.303)	0.406 (0.407)	0.380 (0.380)	0.422	0.604	0.389
MaskGIT-medium	0.254 (0.254)	0.347 (0.347)	0.330 (0.330)	0.353	0.516	0.341
ImageNet						
Strip pool	0.686 (0.711)	0.816 (0.823)	0.737 (0.756)	0.891	1.075	0.760
CNN-S (3 seeds)	0.588 (0.588)	0.704 (0.703)	0.631 (0.631)	0.761	0.987	0.640
MaskGIT-tiny	0.573 (0.573)	0.674 (0.674)	0.589 (0.589)	0.771	0.912	0.598
MaskGIT-medium	0.528 (0.527)	0.617 (0.618)	0.539 (0.539)	0.683	0.805	0.551
NLCD						
Strip pool	0.565 (0.578)	0.564 (0.577)	0.491 (0.507)	0.876	0.876	0.523
CNN-S (3 seeds)	0.497 (0.497)	0.495 (0.495)	0.438 (0.438)	0.974	0.977	0.453
MaskGIT-tiny	0.492 (0.494)	0.491 (0.493)	0.433 (0.433)	0.731	0.729	0.444
MaskGIT-medium	0.464 (0.464)	0.462 (0.462)	0.403 (0.403)	0.697	0.690	0.418

CNN-S values are means over three separately evaluated training seeds, not ensemble scores. For each model, seed and mask, $\widehat{\tau}$ minimizes ordered NLL on 300 development tiles over $\tau \in \{0.50, 0.55, \ldots, 2.50\}$. Selected temperatures are retained even when their test NLL is slightly higher than at $\tau = 1$. All ordered scores use a fixed raster order and true preceding hidden tokens. First-pass scores use only the original observations at $\tau = 1$. The strip row excludes higher-order contexts. MaskGIT references follow Chang et al. (2022); all values are measured under our token-grid protocol.