WTN: LOW-RANK INVERTIBLE ADAPTERS FOR LOW-BIT EMBEDDING CACHE MIGRATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Updating low-bit embedding caches for new tasks can disrupt services that depend on earlier representations. We introduce **W**asserstein **T**ransport **N**etworks (WTN), a constrained family of low-rank convex-gradient residual maps designed to support both task adaptation and historical recovery. WTN learns a flexible set of transport directions under joint constraints that ensure a fixed residual Lipschitz bound, global bijectivity, and stable inversion. The non-expansive inverse allows the same learned maps used for adaptation to recover historical representations, with recovery errors bounded by accumulated quantization and numerical inversion errors. As a result, WTN requires only the latest cache rather than storing historical representations. Across four hierarchical vision datasets and multiple 3-bit migrations, WTN maintains strong latest-task performance while limiting degradation on historical tasks and preserving the neighborhood structure of older representations.

1 INTRODUCTION

Machine-learning systems often store the feature vectors (embeddings) produced by an encoder so that downstream services can reuse them without re-encoding the original inputs. After a model or task upgrade, a representation better suited to the current objective may be incompatible with consumers built for an earlier version. Re-extracting all features, retaining every cache, or retraining every consumer becomes costly as data and version histories grow (Shen et al., 2020; Hu et al., 2022). Production systems also report that capacity limits can force older embedding versions to be retired before every downstream consumer has migrated (Liu et al., 2025).

As class hierarchies become more fine-grained, existing feature caches may need to support finer-grained prediction tasks (Lin et al., 2022). Low-bit embeddings reduce industrial retrieval costs (Gan et al., 2023), but retaining a cache after every update would still make storage grow with the number of versions. Recomputing features is not always possible: BiCT studies how to update stored embeddings when raw gallery inputs cannot be revisited (Su et al., 2022). Meanwhile, existing services may still depend on earlier representations. BC-Aligner supports these services by mapping the latest embeddings back to earlier versions (Hu et al., 2022). As updates accumulate, historical services may require representations several versions removed from the latest cache. We study how to support both current and historical tasks while retaining only the latest low-bit cache. Each generation adapts and requantizes the existing cache within the same bit budget, then replaces it with the new version. The maps, quantizers, and task heads from each generation are retained. Current tasks read the latest cache directly, while historical services recover earlier representations.

The goal is to adapt cached features for a new task while recovering earlier representations accurately and at a practical computational cost. BCT, BC-Aligner, and HBCT use explicit compatibility objectives, but their forward and historical-service paths are not the two directions of one bijection (Shen et al., 2020; Hu et al., 2022; Bui et al., 2025). Repeated low-bit migration requires controlling error propagation through successive inverses while keeping recovery efficient. We compare invertible alternatives, including iResNet, CP-Flow, BiLipNet, and BLNN, under this setting (Behrmann et al., 2019; Huang et al., 2021; Wang et al., 2024; Kinoshita & Toyoizumi, 2024).

Our contributions are:

- We introduce WTN, a constrained family of low-rank convex-gradient residual maps for task adaptation and historical recovery from a single low-bit cache.

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.

- WTN is globally bijective, with a non-expansive inverse and a rank-independent residual Lipschitz bound of $\tau/4$. For $0 < \tau < 4$, fixed-point inversion is contractive. With exact inversion, recovery error is bounded by the sum of quantization errors across migrations.
- Experiments on four hierarchical datasets show that WTN adapts to new tasks with limited historical accuracy loss and preserves historical neighborhoods across repeated 3-bit migrations. Ablations identify the effects of individual constraints.

2 Related Work

2.1 Backward-Compatible Representation Learning and Evolving Class Ontologies

Backward-compatible representation learning aims to keep legacy consumers usable after a model upgrade. BCT uses the legacy classifier to constrain the new model so that old and new embeddings can interact directly (Shen et al., 2020). HBCT preserves compatibility across generations through hyperbolic ordering and contrastive alignment (Bui et al., 2025). BC-Aligner jointly trains a new embedding model and a separate backward map to the preceding version, composing these maps to approximate earlier representations (Hu et al., 2022). WTN uses the same parameters for task adaptation and historical recovery. LECO studies the progressive refinement of class ontologies from coarse to fine (Lin et al., 2022). Our experiments follow this form of task evolution.

2.2 Convex-Gradient and Monotone Networks

ICNNs parameterize convex scalar potentials whose input gradients are monotone maps (Amos et al., 2017). ICGNs directly construct convex gradients through integrated Gram Jacobians (Richter-Powell et al., 2021). Monotone Gradient Networks and the monotone variants of GradNets directly parameterize gradients of convex functions (Chaudhari et al., 2023; 2025). Within this family, WTN combines a rank-independent residual Lipschitz bound with a non-expansive exact inverse and contractive fixed-point inversion.

2.3 Invertible Networks and Lipschitz-Stable Structures

iResNet ensures invertibility by constraining its residual Lipschitz constant to $L < 1$ (Behrmann et al., 2019). For $L > 0$, its inverse Lipschitz bound $1/(1 - L) > 1$ does not rule out perturbation amplification. CP-Flow parameterizes a map as the gradient of a strongly convex potential and computes its inverse by solving the associated convex optimization problem (Huang et al., 2021). CPL constructs 1-Lipschitz convex-potential layers from a dynamical-systems perspective, with forward robustness as its primary application (Meunier et al., 2022). BiLipNet controls forward and inverse Lipschitz constants through strongly monotone layers, orthogonal layers, and incremental quadratic constraints; it computes its inverse by operator splitting (Wang et al., 2024). BLNN parameterizes a globally bi-Lipschitz map using convex neural networks and Legendre-Fenchel duality (Kinoshita & Toyoizumi, 2024). For WTN, both a forward pass and each fixed-point iteration cost $O(rd)$ per vector, using the same learned parameters.

3 Method

3.1 Multi-Generation Low-Bit Cache Migration

Let E denote the frozen encoder and let $u(x) = E(x) \in \mathbb{R}^d$ be its base feature. Generation 0 jointly trains a dimension-preserving supervised stem $S_0 : \mathbb{R}^d \to \mathbb{R}^d$ and a task head C_0 on the first hierarchy level. After validation selection, S_0 and C_0 are frozen, and the initial floating-point representation is

$$h_0(x) = S_0\big(u(x)\big) \in \mathbb{R}^d. \tag{1}$$

A low-bit quantizer Q_0, calibrated using training data only, defines the initial deployed cache

$$c_0(x) = Q_0\big(h_0(x)\big). \tag{2}$$

The generation-0 service reads the deployed representation, so $\widehat{y}_0 = C_0(c_0)$. We use Q_k for a calibrated and frozen quantize-dequantize operator and $c_k(x) \in \mathbb{R}^d$ for the decoded numerical vector passed to subsequent models.

Suppose the system undergoes K upgrades. At each generation $k \geq 1$, migration uses the preceding decoded cache $c_{k-1}(x)$ as its sole representation input and the current-level label $y^{(k)}$ as supervision. Sample identities and data splits remain fixed across generations; only the hierarchical target level changes. Generation k trains a dimension-preserving map $T_k : \mathbb{R}^d \to \mathbb{R}^d$ together with a task head C_k. After validation-based selection, the next deployed cache is

$$c_k(x) = Q_k\big(T_k\big(c_{k-1}(x)\big)\big)\,, \tag{3}$$

which gives the recursive chain

$$c_0 \xrightarrow{T_1,Q_1} c_1 \xrightarrow{T_2,Q_2} \cdots \xrightarrow{T_K,Q_K} c_K. \tag{4}$$

Training at generation k uses current-task supervision on pre-quantization representations, while checkpoint selection evaluates the quantized validation path. The selected quantizer is recalibrated on transformed training representations and frozen before test evaluation.

The latest task reads c_K directly, giving $\widehat{y}_K = C_K(c_K)$. A historical head C_j, frozen after generation j, is served by reversing the subsequent transformations:

$$\widehat{c}_j^{(K)} = T_{j+1}^{-1} \circ T_{j+2}^{-1} \circ \cdots \circ T_K^{-1}(c_K), \qquad \widehat{y}_j = C_j\Big(\widehat{c}_j^{(K)}\Big)\,. \tag{5}$$

Here C_j includes the LayerNorm, linear classifier, and bias deployed at generation j.

Historical compatibility is measured relative to the cache that was actually deployed. Let $\mathcal{A}_j(v)$ denote the accuracy of C_j on representation v. After K migrations,

$$\Delta_j^{(K)} = \mathcal{A}_j(c_j) - \mathcal{A}_j\Big(\widehat{c}_j^{(K)}\Big)\,, \qquad \Delta_{\max}^{(K)} = \max_{0 \leq j < K} \Delta_j^{(K)}. \tag{6}$$

Positive values indicate degradation; negative values indicate that the recovered representation performs better under the frozen historical head. For N samples quantized at b bits per coordinate, one retained cache uses Ndb bits under ideal bit packing; all $K+1$ versions use $(K+1)Ndb$ bits. Appendix D accounts for the per-generation model and quantizer state separately.

3.2 Constrained Low-Rank Convex-Gradient Parameterization

Here and in Section 3.3, $x \in \mathbb{R}^d$ denotes an input representation. Generation k uses r_k normalized directions $w_{k,\ell} \in \mathbb{R}^d$ with $\|w_{k,\ell}\|_2 \leq 1$. Each direction has a positive weight $a_{k,\ell}$ and a bias $b_{k,\ell}$, with $\sum_\ell a_{k,\ell} = 1$. With translation $t_k \in \mathbb{R}^d$ and residual scale $\tau_k > 0$, define

$$\psi_k(x) = \frac{1}{2}\|x\|_2^2 + t_k^\top x + \tau_k \sum_{\ell=1}^{r_k} a_{k,\ell}\,\mathrm{softplus}\big(w_{k,\ell}^\top x + b_{k,\ell}\big)\,. \tag{7}$$

Its gradient gives the WTN map

$$T_k(x) = x + t_k + \tau_k \sum_{\ell=1}^{r_k} a_{k,\ell}\sigma\big(w_{k,\ell}^\top x + b_{k,\ell}\big)\, w_{k,\ell}, \tag{8}$$

where σ is the logistic sigmoid. Figure 1(b) illustrates the translation and directional displacements. Let $W_k \in \mathbb{R}^{r_k \times d}$ stack the directions row-wise, and let $a_k, b_k \in \mathbb{R}^{r_k}$ collect their weights and biases. Trainable logits $\rho_k \in \mathbb{R}^{r_k}$ define the shared weights $a_k = \mathrm{softmax}(\rho_k)$; the gates $\sigma(p)$ depend on the input through $p = W_k x + b_k$. The input and output remain in $\mathbb{R}^d$, while the sample-dependent displacement is confined to the learned directional subspace:

$$T_k(x) - x - t_k \in \mathrm{span}\{w_{k,1}, \ldots, w_{k,r_k}\}, \qquad \mathrm{rank}(J_{T_k - I}(x)) \leq \mathrm{rank}(W_k) \leq r_k. \tag{9}$$

Thus low rank describes the nonlinear displacement and its Jacobian. Increasing the number of directions r_k raises the maximum available directional rank, while the realized span depends on the learned directions. When $r_k \geq d$ and W_k spans $\mathbb{R}^d$, the residual Jacobian can be full rank; smaller

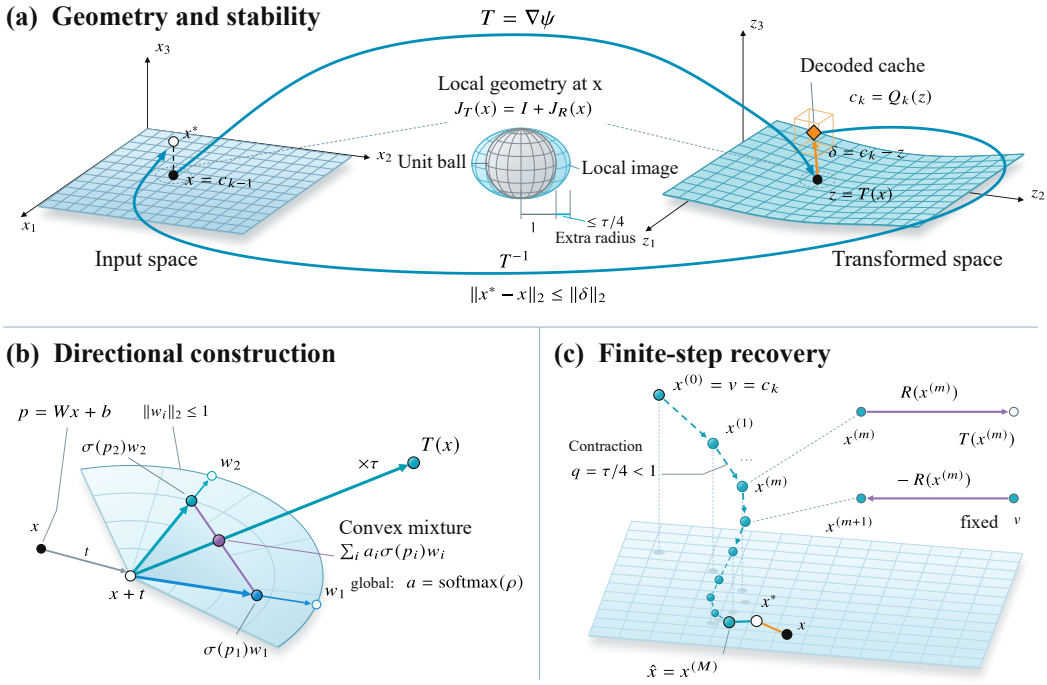


Figure 1: **WTN geometry and recovery.** (a) Forward migration, quantization, and non-expansive exact inversion. (b) Residual construction from sigmoid-gated directions; the sector marks unit length. (c) Fixed-point recovery distinguishes the finite-step estimate $\widehat{x}$, exact inverse x^*, and original input x. Panels are schematic, not to scale; generation indices on WTN maps and parameters are omitted.

values reduce model state and computation. The simplex constraint keeps the $\tau_k/4$ residual certificate independent of r_k.

We evaluate a fixed-scale setting and a bounded learnable-scale setting, both enforcing $0 < \tau_k < 4$. The raw parameterization, batch evaluation, and scale construction are given in Appendix A. A fixed-scale block contains $r_k d + 2r_k + d$ trainable parameters. For a batch of B vectors, a forward pass or one fixed-point step costs approximately $2Br_k d$ multiply-accumulate operations. Because $T_k = \nabla\psi_k$ with ψ_k strongly convex, it also admits the induced quadratic-cost optimal-transport interpretation stated in Appendix A.

3.3 Joint Invertibility and Solver Certificate

Write $R_k(x) = T_k(x) - x$. Its Jacobian is

$$J_{R_k}(x) = \tau_k \sum_{\ell=1}^{r_k} a_{k,\ell}\sigma'\left(w_{k,\ell}^\top x + b_{k,\ell}\right) w_{k,\ell} w_{k,\ell}^\top. \tag{10}$$

Since $a_{k,\ell} > 0$, $0 \leq \sigma'(s) \leq 1/4$, $\|w_{k,\ell}\|_2 \leq 1$, and $\sum_\ell a_{k,\ell} = 1$,

$$0 \preceq J_{R_k}(x) \preceq \frac{\tau_k}{4} I, \qquad \|J_{R_k}(x)\|_2 \leq \frac{\tau_k}{4}. \tag{11}$$

Hence $J_{T_k}(x) = I + J_{R_k}(x) \succeq I$, so T_k is 1-strongly monotone. Together with the supercoercivity of ψ_k, this proves that T_k is a global bijection. Moreover,

$$\left\|T_k^{-1}(u) - T_k^{-1}(v)\right\|_2 \leq \|u - v\|_2. \tag{12}$$

The full argument is given in Appendix A.

The condition $\tau_k < 4$ is used for the deployment solver rather than for bijectivity. Given $v = T_k(x^*)$, initialize $x^{(0)} = v$ and iterate

$$x^{(m+1)} = v - R_k\left(x^{(m)}\right). \tag{13}$$

The update in Figure 1(c) has contraction factor at most $q_k = \tau_k/4 < 1$, giving

$$\|x^{(m)} - x^*\|_2 \leq \left(\frac{\tau_k}{4}\right)^m \|x^{(0)} - x^*\|_2. \tag{14}$$

After M iterations, the estimate is $\widehat{x} = x^{(M)}$, with numerical error decreasing geometrically in M.

3.4 Multi-Generation Quantization-Error Propagation

Without intermediate quantization, the inverse of $S_{j\to K} = T_K \circ \cdots \circ T_{j+1}$ is a composition of 1-Lipschitz maps and is therefore non-expansive:

$$\left\| S_{j\to K}^{-1}(u) - S_{j\to K}^{-1}(v) \right\|_2 \leq \|u - v\|_2. \tag{15}$$

We fix K throughout the derivation and omit it from superscripts on recovered representations. In the deployed chain, for a fixed sample x, let $z_k(x) = T_k(c_{k-1}(x))$ and $\delta_k(x) = c_k(x) - z_k(x) = Q_k(z_k(x)) - z_k(x)$. Starting from $\widehat{c}_K(x) = c_K(x)$, define exact recovery by $\widehat{c}_{k-1}(x) = T_k^{-1}(\widehat{c}_k(x))$ and $e_k(x) = \|\widehat{c}_k(x) - c_k(x)\|_2$. For inverse Lipschitz bounds L_k, recursive expansion gives

$$e_j(x) \leq \sum_{m=j+1}^{K} \left(\prod_{k=j+1}^{m} L_k \right) \|\delta_m(x)\|_2. \tag{16}$$

For WTN, $L_k \leq 1$, and therefore

$$e_j(x) \leq \sum_{m=j+1}^{K} \|\delta_m(x)\|_2. \tag{17}$$

Deployment uses a finite-step approximation $\widetilde{T}_k^{-1}$; we now omit the fixed sample argument x. Starting from $\widetilde{c}_K = c_K$, set $\widetilde{c}_{k-1} = \widetilde{T}_k^{-1}(\widetilde{c}_k)$. Define the numerical inversion error at each input encountered during recovery as

$$\eta_k = \left\| \widetilde{T}_k^{-1}(\widetilde{c}_k) - T_k^{-1}(\widetilde{c}_k) \right\|_2. \tag{18}$$

For WTN, the recovery error $\widetilde{e}_j = \|\widetilde{c}_j - c_j\|_2$ satisfies

$$\widetilde{e}_j \leq \sum_{m=j+1}^{K} \|\delta_m\|_2 + \sum_{m=j+1}^{K} \eta_m. \tag{19}$$

Quantization and numerical inversion errors accumulate additively in the recovery bound; inverse non-expansiveness prevents their amplification. Appendix A gives the derivation and the corresponding empirical RMSE bound.

4 Experiments

4.1 Experimental Protocol

Our hierarchical benchmark studies recursive cache migration on iNaturalist 2019 (Van Horn et al., 2018), FungiTastic (Picek et al., 2025), a filtered split derived from the Shopify Product Catalogue (Shopify, 2025), and BIOSCAN-5M (Gharaee et al., 2024). Their natural class hierarchies define four to six successive upgrades. A frozen DINOv2-S encoder (Oquab et al., 2024) provides 384-dimensional base CLS features, and caches are quantized to three bits per coordinate. All methods start from the same generation-0 cache. Training and structural hyperparameters are selected on validation data and then held fixed.

We compare WTN with other invertible adapters, Repeated Raw-Q, BC-Aligner, HBCT, and Keep-All. Repeated Raw-Q reuses fixed coordinates; Keep-All retains earlier caches. BC-Aligner learns backward maps, while HBCT serves the historical heads included in training. The primary task metrics are final-generation accuracy and the maximum cumulative historical-task degradation in Equation (6). Means and sample SDs summarize independently trained models. We time recovery of

the oldest version on an exclusively allocated NVIDIA L20 using decoded tensors already resident on the GPU. Timing excludes bit unpacking and host-to-device transfer. For historical retrieval, we evaluate WTN checkpoints selected on validation data with 1,024 test queries and a fixed historical validation gallery. We evaluate the retrieval baselines using frozen checkpoints. Appendices B and D report retrieval results and computational costs. Section 4.5 evaluates two successive encoder upgrades with mixed galleries.

4.2 Multi-Generation Cache Migration

Table 1 reports representative results after the final migration. The WTN row uses the residual-scale setting selected on validation data before test evaluation: learnable scale on iNaturalist, FungiTastic, and BIOSCAN-5M, and fixed scale on Shopify. Results for both WTN scale parameterizations and the remaining configurations appear in Appendix B.

Table 1: Representative results after the final migration. Rows report latest-task accuracy (%, ↑) and maximum cumulative historical-task degradation (percentage points, ↓), as mean ± sample standard deviation. Boldface marks the highest mean latest-task accuracy in each dataset. WTN uses the residual-scale setting selected on validation data.

Method	Metric	iNaturalist 2019	Shopify	FungiTastic	BIOSCAN-5M
Repeated Raw-Q	Accuracy (%)	63.452 ± 0.710	39.497 ± 0.778	50.906 ± 0.309	44.918 ± 0.103
	Max. drop (pp)	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
WTN	Accuracy (%)	$\mathbf{63.578} \pm 0.488$	39.808 ± 0.702	$\mathbf{51.207} \pm 0.266$	$\mathbf{45.407} \pm 0.137$
	Max. drop (pp)	0.528 ± 0.249	0.323 ± 0.201	1.027 ± 0.130	0.881 ± 0.482
iResNet (Behrmann et al., 2019)	Accuracy (%)	62.660 ± 0.501	$\mathbf{39.952} \pm 0.748$	50.447 ± 0.191	43.495 ± 0.267
	Max. drop (pp)	2.079 ± 0.772	3.052 ± 0.745	2.434 ± 0.477	5.193 ± 2.227
BiLipNet (Wang et al., 2024)	Accuracy (%)	58.033 ± 0.576	37.163 ± 0.747	46.438 ± 0.132	40.729 ± 0.350
	Max. drop (pp)	2.297 ± 0.511	5.458 ± 2.917	48.306 ± 6.223	48.228 ± 8.068
CP-Flow (Huang et al., 2021)	Accuracy (%)	59.109 ± 0.386	39.856 ± 0.635	44.502 ± 0.614	42.670 ± 0.507
	Max. drop (pp)	1.142 ± 0.771	0.287 ± 0.375	4.645 ± 2.908	5.261 ± 3.273
CPL (Meunier et al., 2022)	Accuracy (%)	58.944 ± 0.798	38.049 ± 0.810	44.812 ± 0.234	38.376 ± 0.439
	Max. drop (pp)	30.330 ± 5.281	5.087 ± 2.243	59.090 ± 7.390	12.697 ± 2.534
BLNN (Kinoshita & Toyoizumi, 2024)	Accuracy (%)	63.195 ± 0.515	39.665 ± 0.883	50.875 ± 0.323	44.502 ± 0.229
	Max. drop (pp)	0.825 ± 0.170	0.395 ± 0.353	10.230 ± 2.279	1.192 ± 0.415
BC-Aligner (Hu et al., 2022)	Accuracy (%)	55.531 ± 1.092	27.528 ± 0.236	46.527 ± 0.141	38.886 ± 0.503
	Max. drop (pp)	3.875 ± 2.437	1.388 ± 0.716	15.194 ± 6.165	13.292 ± 4.580
HBCT (Bui et al., 2025)	Accuracy (%)	62.719 ± 1.672	39.366 ± 0.817	50.742 ± 0.537	44.868 ± 0.099
	Max. drop (pp)	0.046 ± 0.068	0.838 ± 0.579	0.476 ± 0.600	0.043 ± 0.025
Keep-All	Accuracy (%)	62.139 ± 0.540	38.372 ± 1.042	49.456 ± 0.186	43.547 ± 0.098
	Max. drop (pp)	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000

In Table 1, WTN has the highest mean latest-task accuracy on iNaturalist, FungiTastic, and BIOSCAN-5M and is 0.144 percentage points below iResNet on Shopify. Its improvement over Repeated Raw-Q is modest, ranging from 0.126 to 0.489 points. The clearer difference is historical recovery: among the structural-inverse baselines, WTN has the lowest maximum degradation on iNaturalist, FungiTastic, and BIOSCAN-5M, and is 0.036 points above CP-Flow on Shopify.

Controlled storage oracle. To isolate cache retention, the WTN Keep-All oracle shares the checkpoints, heads, quantizers, and latest cache of single-cache WTN while retaining every intermediate cache. The Keep-All baseline in Table 1 is trained independently. Latest-task predictions are identical between the oracle and single-cache WTN; the oracle's historical prediction agreement is one and recovery RMSE is zero. Single-cache WTN reduces storage from five to seven cache copies (720-1008 bytes per vector) to one (144 bytes), with maximum historical degradation of 0.323-1.027 percentage points.

Recovery over version distance and legacy retrieval. Figure 2 shows recovery RMSE, historical-task degradation, and neighborhood preservation at each version distance. Oldest-version exact top-10 overlap for validation-selected WTN is 0.9122 ± 0.0254, 0.9675 ± 0.0056, 0.9006 ± 0.0079,

and 0.9498 ± 0.0053 on iNaturalist, Shopify, FungiTastic, and BIOSCAN-5M, respectively (mean $\pm$ sample SD). Complete curves and cosine-score RMSE are in Appendix B.

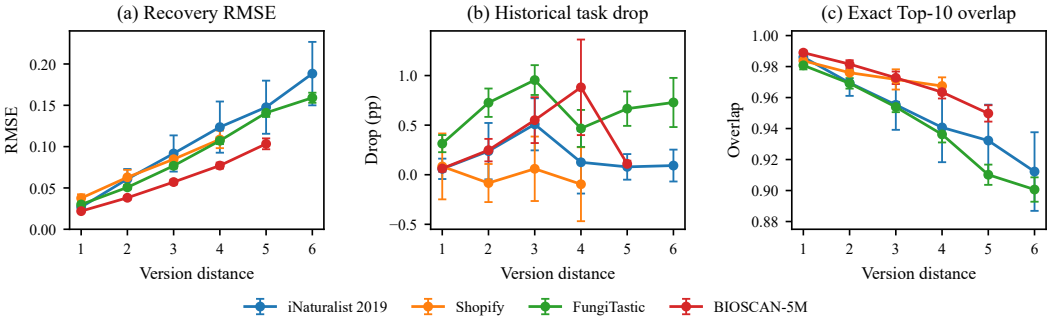


Figure 2: **Historical recovery across version distance.** (a) RMSE against the deployed, decoded historical cache. (b) Historical accuracy loss in percentage points. (c) Exact Top-10 overlap. Original and recovered queries search the same fixed historical validation gallery, using 1,024 test queries. Negative accuracy losses indicate improvements. All panels show means and sample SDs.

Model-side recovery cost. WTN serves the oldest historical task in 3.198-6.206 ms (Figure 3).

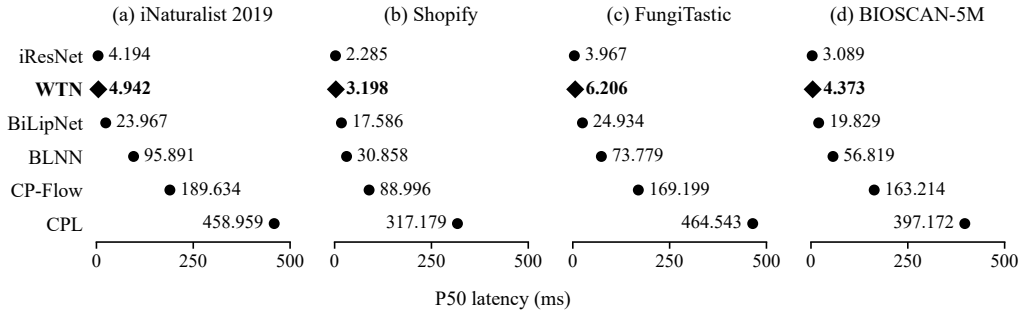


Figure 3: Model-side latency for serving the oldest historical task with structural-inverse methods. Points show P50 latency at batch size one on an NVIDIA L20, using decoded GPU-resident tensors. All panels use the same linear scale. Appendix D.3 reports complete timings.

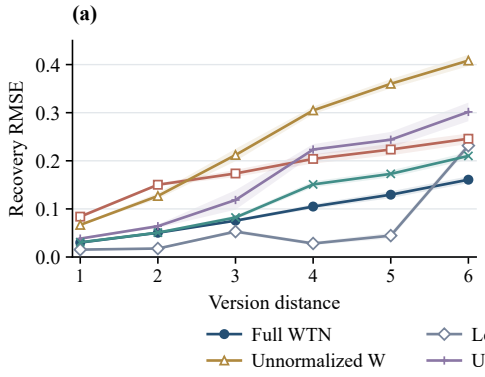
WTN is faster than BiLipNet, BLNN, CP-Flow, and CPL on every dataset. iResNet is faster at 2.285-4.194 ms, but WTN has lower mean historical degradation on all four datasets (Table 1). Appendix D gives complete timings and storage costs.

4.3 Structural Constraints and Historical Recovery

We examine how WTN's constraints affect task adaptation and historical recovery. Figure 4 shows FungiTastic, including a variant with fixed alternating coefficient signs. Appendix C.1 defines this variant and reports the matched single-factor comparisons across four datasets.

Full WTN has the lowest mean recovery RMSE at the oldest target version in Figure 4(a). Removing direction normalization reduces latest-task accuracy by 2.106, 3.834, and 1.646 percentage points on iNaturalist, FungiTastic, and BIOSCAN-5M, respectively. All three differences remain significant after Holm correction across 40 tests. On FungiTastic, it also increases historical degradation by 14.042 points (adjusted $p = 0.0060$), despite retaining a non-expansive exact inverse.

The fixed-sign variant increases historical degradation by 10.87 points while reducing latest-task accuracy by 0.38 points (Figure 4(b)). By contrast, the matched variant initialized with positive coefficients learns little negative mass and remains close to full WTN (Appendix C.1).



(b)

Method	Δ Accuracy (pp)	Δ Max. drop (pp)
Full WTN	0.00	0.00
Learnable α	-0.41±0.09	+0.89±1.15
Signed-WTN (learnable τ)	-0.38±0.10	+10.87±2.49
Unnormalized W	-3.83±0.40	+14.04±2.26
Unnormalized a	-3.10±0.56	+3.08±1.57
Unbounded τ	-0.20±0.08	+1.04±0.56

Figure 4: Structural ablations on FungiTastic. (a) Recovery RMSE from the final cache versus version distance $K - j$. (b) Paired differences in latest-task accuracy and maximum degradation across historical tasks recovered from the final cache (variant minus Full WTN, pp). Results show means and sample SDs. Appendix C.1 defines the variants and reports the four-dataset comparisons.

Perturbation amplification. We measure how small perturbations propagate during recovery from the final cache to generation 0. Table 2 reports the largest analytic gain over 512 sampled directions for each frozen checkpoint.

Table 2: Maximum sampled gain during recovery from the final cache to generation 0. For each frozen checkpoint, values are the maximum analytic gain over 512 sampled directions. WTN uses a fixed residual scale.

Method	iNaturalist 2019	Shopify	FungiTastic	BIOSCAN-5M
WTN (fixed scale)	0.9974	0.9969	0.9977	0.9973
Signed-WTN (fixed τ)	1.0713	1.0092	1.1033	1.0271
Free-signed WTN	1.0835	1.0092	1.1964	1.0728
iResNet	1.1407	1.0086	1.0647	1.0676
BiLipNet	0.9830	0.9933	0.9821	0.9865
BLNN	0.0197	0.0709	0.0205	0.0413
CP-Flow	4.4618	1.1515	7.5306	2.9495
CPL	12.6951	4.9961	64.0773	14.6873

Fixed-scale WTN stays below one on all four datasets, whereas iResNet and both signed variants exceed one. CP-Flow and CPL show stronger amplification, reaching gains of 7.53 and 64.08 on FungiTastic. BiLipNet and BLNN also remain below one in these probes. Appendix C.3 compares the analytic gains with finite differences and discusses unsampled directions.

On FungiTastic, CPL reaches a maximum clean reconstruction RMSE of 1.38×10^{-6} with 200 inverse iterations per stage (Appendix C.4). Yet accuracy on the generation-0 kingdom classification task drops by 48.477 percentage points after recovery from the quantized cache.

4.4 Image Copy Detection with a Legacy Verifier

We use DISC21 (Douze et al., 2021) to test an updated retrieval service that must retain a legacy copy verifier. The evaluation uses 50,000 reference images and 1,058 queries, including 529 with a matching reference. After two 3-bit migrations, the service retrieves 100 candidates from the latest cache, recovers the query and candidate representations, and applies the unchanged generation-0 cosine threshold. The service retains only the latest gallery cache. Appendix E.1 reports service metrics and further examples.

Recall@100 rises from 71.834% to 73.422%, while micro-AP falls from 35.053% to 34.408%. The inverse service retains 99.0% of the old service's 60 correct returns; recall over all positive queries changes from 11.342% to 11.229%. Both true references in Figure 5 rank first in the latest cache, but the unchanged verifier accepts (a) and rejects (b).

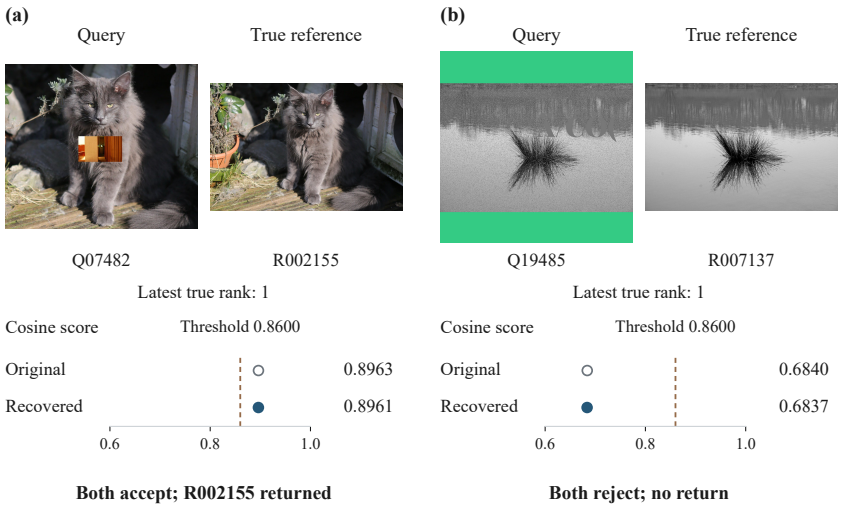


Figure 5: DISC21 copy detection. Cosine scores use decoded generation-0 query-reference vectors (Original) or their recovered counterparts (Recovered). Dashed lines mark the fixed verifier threshold.

4.5 Encoder Upgrades

We test two DINOv2 upgrades (S $\rightarrow$ B $\rightarrow$ L) on Aircraft (Maji et al., 2013) and iNaturalist-Loc500 (Cole et al., 2022) in a shared 1024-dimensional space. Frozen Procrustes bridges and orthogonal extensions retain features from DINOv2-B and DINOv2-L; the 384-dimensional starting cache is zero-padded (Appendix E.2).

Table 3: Latest-task accuracy and mixed-gallery mAP@10 after two encoder upgrades at 1024 dimensions (test set; mean $\pm$ sample SD). Bold marks the highest unrounded mean in each column.

	Aircraft		iNaturalist-Loc500	
Method	Accuracy (%)	mAP@10	Accuracy (%)	mAP@10
No migration	61.128±0.598	0.1471±0.0000	76.482±0.164	0.3691±0.0000
Procrustes	56.472±0.462	**0.1536**±0.0000	76.608±0.098	0.3817±0.0000
Nonlinear low-rank	62.544±0.240	0.1492±0.0006	75.883±0.076	0.3833±0.0016
iResNet	61.896±0.355	0.1503±0.0005	76.050±0.095	**0.3833**±0.0006
WTN	**65.065**±0.383	0.1481±0.0011	**76.819**±0.146	0.3707±0.0004

WTN has the highest mean latest-task accuracy on both datasets (Table 3). Procrustes and iResNet have the highest mean mAP@10 on Aircraft and iNaturalist-Loc500, respectively. Appendix E.2 gives the configurations, search settings, and historical recovery results.

5 Conclusion

WTN adapts quantized caches and recovers historical representations through the same learned maps, retaining only the latest cache. Its non-expansive inverse controls error propagation, while historical access trades storage for computation on demand. Across repeated 3-bit migrations, WTN maintains competitive latest-task accuracy with limited historical degradation and preserves historical neighborhoods. The recovery experiments distinguish numerical inversion accuracy from stability under quantization. The copy-detection case shows that an updated cache can serve an unchanged verifier, although compatibility alone does not ensure better detection performance.

AI USE STATEMENT

Generative AI tools assisted with English writing and language editing, as well as retrieval and discovery of related work. The authors independently reviewed the resulting text, verified the relevant citations, and take responsibility for all content.

REFERENCES

Brandon Amos, Lei Xu, and J. Zico Kolter. Input convex neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 146-155. PMLR, 2017.

Jens Behrmann, Will Grathwohl, Ricky T. Q. Chen, David Duvenaud, and Jörn-Henrik Jacobsen. Invertible residual networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 573-582. PMLR, 2019.

Yann Brenier. Polar factorization and monotone rearrangement of vector-valued functions. *Communications on Pure and Applied Mathematics*, 44(4):375-417, 1991. doi: 10.1002/cpa.3160440402.

Ngoc Bui, Menglin Yang, Runjin Chen, Leonardo Neves, Mingxuan Ju, Zhitao Ying, Neil Shah, and Tong Zhao. Learning along the arrow of time: Hyperbolic geometry for backward-compatible representation learning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 5842-5855. PMLR, 2025.

Shreyas Chaudhari, Srinivasa Pranav, and José M. F. Moura. Learning gradients of convex functions with monotone gradient networks. In *2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1-5. IEEE, 2023. doi: 10.1109/ICASSP49357.2023.10097266.

Shreyas Chaudhari, Srinivasa Pranav, and José M. F. Moura. Gradient networks. *IEEE Transactions on Signal Processing*, 73:324-339, 2025. doi: 10.1109/TSP.2024.3496692.

Elijah Cole, Kimberly Wilber, Grant Van Horn, Xuan Yang, Marco Fornoni, Pietro Perona, Serge Belongie, Andrew Howard, and Oisin Mac Aodha. On label granularity and object localization. In *European Conference on Computer Vision*, pp. 604-620. Springer, 2022.

Matthijs Douze, Giorgos Tolias, Ed Pizzi, Zoë Papakipos, Lowik Chanussot, Filip Radenovic, Tomas Jenicek, Maxim Maximov, Laura Leal-Taixé, Ismail Elezi, Ondřej Chum, and Cristian Canton Ferrer. The 2021 image similarity dataset and challenge. *arXiv preprint arXiv:2106.09672*, 2021. URL https://arxiv.org/abs/2106.09672.

Yukang Gan, Yixiao Ge, Chang Zhou, Shupeng Su, Zhouchuan Xu, Xuyuan Xu, Quanchao Hui, Xiang Chen, Yexin Wang, and Ying Shan. Binary embedding-based retrieval at Tencent. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 4056-4067. ACM, 2023. doi: 10.1145/3580305.3599782.

Zahra Gharaee, Scott C. Lowe, ZeMing Gong, Pablo Millan Arias, Nicholas Pellegrino, Austin T. Wang, Joakim Bruslund Haurum, Iuliia Zarubiieva, Lila Kari, Dirk Steinke, Graham W. Taylor, Paul Fieguth, and Angel X. Chang. BIOSCAN-5M: A multimodal dataset for insect biodiversity. In *Advances in Neural Information Processing Systems*, volume 37, pp. 36285-36313. Curran Associates, Inc., 2024. doi: 10.52202/079017-1144.

Weihua Hu, Rajas Bansal, Kaidi Cao, Nikhil Rao, Karthik Subbian, and Jure Leskovec. Learning backward compatible embeddings. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 3018-3028. ACM, 2022. doi: 10.1145/3534678.3539194.

Chin-Wei Huang, Ricky T. Q. Chen, Christos Tsirigotis, and Aaron Courville. Convex potential flows: Universal probability distributions with optimal transport and convex optimization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=te7PVH1sPxJ.

Yuri Kinoshita and Taro Toyoizumi. A provable control of sensitivity of neural networks through a direct parameterization of the overall bi-Lipschitzness. In *Advances in Neural Information Processing Systems*, volume 37, pp. 85241-85309. Curran Associates, Inc., 2024. doi: 10.52202/079017-2707.

Zhiqiu Lin, Deepak Pathak, Yu-Xiong Wang, Deva Ramanan, and Shu Kong. Continual learning with evolving class ontologies. In *Advances in Neural Information Processing Systems*, volume 35, pp. 7671-7684. Curran Associates, Inc., 2022. doi: 10.52202/068431-0557.

Ping Liu, Rajat Arora, Xiao Shi, Benjamin Hoan Le, Qianqi Shen, Jianqiang Shen, Chengming Jiang, Nikita Zhiltsov, Priya Bannur, Yidan Zhu, Liming Dong, Haichao Wei, Qi Guo, Luke Simon, Liangjie Hong, and Wenjing Zhang. A scalable and efficient signal integration system for job matching. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pp. 4659-4669. ACM, 2025. doi: 10.1145/3711896.3737185.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. doi: 10.48550/arXiv.1306.5151.

Laurent Meunier, Blaise J. Delattre, Alexandre Araujo, and Alexandre Allauzen. A dynamical system perspective for Lipschitz neural networks. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 15484-15500. PMLR, 2022.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=a68SUt6zFt.

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

Lukas Picek, Klara Janouskova, Vojtech Cermak, and Jiri Matas. FungiTastic: A multi-modal dataset and benchmark for image categorization. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 2037-2047. IEEE, 2025. doi: 10.1109/CVPRW67362.2025.00192.

Jack Richter-Powell, Jonathan Lorraine, and Brandon Amos. Input convex gradient networks. *arXiv preprint arXiv:2111.12187*, 2021. doi: 10.48550/arXiv.2111.12187. Accepted at the NeurIPS 2021 Workshop on Optimal Transport and Machine Learning.

R. Tyrrell Rockafellar. *Convex Analysis*. Princeton University Press, 1970.

Yantao Shen, Yuanjun Xiong, Wei Xia, and Stefano Soatto. Towards backward-compatible representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6367-6376. IEEE, 2020. doi: 10.1109/CVPR42600.2020.00640.

Shopify. The catalogue: Product taxonomy classification benchmark. Hugging Face dataset, 2025. URL https://huggingface.co/datasets/Shopify/product-catalogue.

Shupeng Su, Binjie Zhang, Yixiao Ge, Xuyuan Xu, Yexin Wang, Chun Yuan, and Ying Shan. Privacy-preserving model upgrades with bidirectional compatible training in image retrieval. *arXiv preprint arXiv:2204.13919*, 2022. doi: 10.48550/arXiv.2204.13919.

Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The iNaturalist species classification and detection dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8769-8778, 2018. doi: 10.1109/CVPR.2018.00914.

Ruigang Wang, Krishnamurthy Dj Dvijotham, and Ian Manchester. Monotone, bi-Lipschitz, and Polyak-Łojasiewicz networks. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 50379-50399. PMLR, 2024.

A Theoretical Properties and Proofs

A.1 Parameterization and Residual Bounds

Generation k uses a raw direction matrix $V_k \in \mathbb{R}^{r_k \times d}$, direction biases $b_k \in \mathbb{R}^{r_k}$, coefficient logits $\rho_k \in \mathbb{R}^{r_k}$, a translation $t_k \in \mathbb{R}^d$, and a residual scale τ_k. Direction ℓ is normalized as

$$w_{k,\ell} = \frac{V_{k,\ell:}^\top}{\max(\|V_{k,\ell:}\|_2, \varepsilon_{\text{norm}})}, \tag{20}$$

where $V_{k,\ell:}$ is row ℓ of V_k and $\varepsilon_{\text{norm}} > 0$ prevents division by zero. The weights are $a_k = \text{softmax}(\rho_k)$. Consequently,

$$\|w_{k,\ell}\|_2 \le 1, \qquad a_{k,\ell} > 0, \qquad \sum_{\ell=1}^{r_k} a_{k,\ell} = 1. \tag{21}$$

Let $W_k \in \mathbb{R}^{r_k \times d}$ stack the normalized directions row-wise. For a batch of B input vectors arranged as rows of $X \in \mathbb{R}^{B \times d}$, Equation (8) becomes

$$P = XW_k^\top + \mathbb{1}_B b_k^\top, \tag{22}$$

$$G = \sigma(P) \odot \left(\mathbb{1}_B a_k^\top\right), \tag{23}$$

$$Z = X + \mathbb{1}_B t_k^\top + \tau_k G W_k. \tag{24}$$

Here $\mathbb{1}_B \in \mathbb{R}^B$ is the all-ones column vector, $\odot$ denotes elementwise multiplication, and σ acts elementwise. The projection and gate matrices satisfy $P, G \in \mathbb{R}^{B \times r_k}$, and the output is $Z \in \mathbb{R}^{B \times d}$. The translation is constant in x, and the sample-dependent term lies in the row span of W_k. Hence

$$\text{rank}(J_{T_k - I}(x)) \le \text{rank}(W_k) \le r_k. \tag{25}$$

In the fixed-scale setting, $\tau_k \in (0, 4)$ is selected as a hyperparameter. In the learnable-scale setting, a scalar γ_k determines the scale under a fixed cap $\tau_{\max,k} < 4$:

$$\tau_k = \tau_{\max,k}\sigma(\gamma_k), \qquad 0 < \tau_k < \tau_{\max,k} < 4. \tag{26}$$

Without direction and simplex normalization, the residual bound is

$$\|J_{R_k}(x)\|_2 \le \frac{\tau_k}{4} \sum_{\ell=1}^{r_k} a_{k,\ell} \|w_{k,\ell}\|_2^2. \tag{27}$$

Direction and simplex normalization reduce this bound to $\tau_k/4$, independently of r_k. It is attained when all unit directions coincide and all sigmoid preactivations are zero.

Since $0 < \sigma(s) < 1$, $a_{k,\ell} > 0$, $\sum_\ell a_{k,\ell} = 1$, and $\|w_{k,\ell}\|_2 \le 1$,

$$\begin{aligned} \|T_k(x) - x - t_k\|_2 &\le \tau_k \sum_{\ell=1}^{r_k} a_{k,\ell} \sigma\left(w_{k,\ell}^\top x + b_{k,\ell}\right) \|w_{k,\ell}\|_2 \\ &\le \tau_k. \end{aligned} \tag{28}$$

The bound applies to the nonlinear directional displacement; the translation t_k is parameterized separately from τ_k.

A.2 Global Bijection and Inverse Lipschitz Bound

Equation (10) gives $J_{R_k}(x) \succeq 0$, so

$$J_{T_k}(x) = I + J_{R_k}(x) \succeq I. \tag{29}$$

Integrating the Jacobian along the segment between x and z yields

$$\langle T_k(x) - T_k(z), x - z \rangle \ge \|x - z\|_2^2. \tag{30}$$

Thus T_k is injective. To establish surjectivity, fix $v \in \mathbb{R}^d$ and consider

$$f_v(x) = \psi_k(x) - v^\top x. \tag{31}$$

The quadratic term makes f_v 1-strongly convex, and it dominates the remaining linear-growth terms, so

$$\frac{f_v(x)}{\|x\|_2} \to \infty \qquad \text{as} \qquad \|x\|_2 \to \infty. \tag{32}$$

Hence f_v has a unique minimizer x^*, whose first-order condition is

$$\nabla f_v(x^*) = T_k(x^*) - v = 0. \tag{33}$$

Every v has exactly one preimage, proving that T_k is a global bijection.

Strong monotonicity and the Cauchy-Schwarz inequality give, for $x \neq z$,

$$\|x - z\|_2 \leq \|T_k(x) - T_k(z)\|_2. \tag{34}$$

Together with Equation (11),

$$\|x - z\|_2 \leq \|T_k(x) - T_k(z)\|_2 \leq \left(1 + \frac{\tau_k}{4}\right) \|x - z\|_2. \tag{35}$$

The lower bound is equivalent to Equation (12). Positivity of the residual Jacobian is sufficient for this conclusion; the condition $\tau_k < 4$ is used only by the fixed-point solver.

A.3 Fixed-Point Convergence

Let $v = T_k(x^*) = x^* + R_k(x^*)$ and define $F_v(x) = v - R_k(x)$. Equation (11) gives

$$\|F_v(x) - F_v(z)\|_2 \leq \frac{\tau_k}{4} \|x - z\|_2. \tag{36}$$

When $\tau_k < 4$, Banach's fixed-point theorem gives a unique fixed point and the error bound in Equation (14).

A.4 Composition and Error Propagation

Composition preserves bijectivity and inverse non-expansiveness. The composite map need not be a convex gradient because products of distinct symmetric positive-definite Jacobians need not be symmetric.

For exact recovery, $T_k^{-1}(z_k) = c_{k-1}$. If the inverse of stage k has Lipschitz upper bound L_k, then

$$\begin{aligned} e_{k-1} &= \left\| T_k^{-1}(\widehat{c}_k) - T_k^{-1}(z_k) \right\|_2 && (37) \\ &\leq L_k \|\widehat{c}_k - z_k\|_2 && (38) \\ &\leq L_k \left(e_k + \|\delta_k\|_2\right). && (39) \end{aligned}$$

Starting from $e_K = 0$ and recursively expanding this inequality gives Equation (16). For WTN, every $L_k \leq 1$, which gives Equation (17).

For the finite-step recovery sequence defined in Section 3.4, Equation (18) and the non-expansiveness of T_k^{-1} give

$$\widetilde{e}_{k-1} \leq \eta_k + \widetilde{e}_k + \|\delta_k\|_2. \tag{40}$$

Recursion from $\widetilde{e}_K = 0$ yields Equation (19). These bounds are pointwise in the input sample. For test samples $\{x_i\}_{i=1}^N$, the empirical recovery RMSE reported in Figure 2 is

$$\mathrm{RMSE}_j^{(K)} = \left[\frac{1}{Nd} \sum_{i=1}^{N} \left\| \widetilde{c}_j^{(K)}(x_i) - c_j(x_i) \right\|_2^2 \right]^{1/2}. \tag{41}$$

Define the stage-wise empirical codec and solver terms

$$\begin{aligned} D_m &= \left[\frac{1}{Nd} \sum_{i=1}^{N} \|\delta_m(x_i)\|_2^2 \right]^{1/2}, \\ H_m &= \left[\frac{1}{Nd} \sum_{i=1}^{N} \eta_m(x_i)^2 \right]^{1/2}. \end{aligned} \tag{42}$$

Applying Minkowski's inequality to the pointwise finite-step bound gives

$$\mathrm{RMSE}_j^{(K)} \le \sum_{m=j+1}^{K} D_m + \sum_{m=j+1}^{K} H_m. \tag{43}$$

Figure 2 plots the realized RMSE on the left-hand side against the deployed decoded cache. Classification and retrieval margins determine how this embedding error affects task decisions.

A.5 Common Nullspaces and the Inverse-Chain Norm

The Jacobian of a WTN inverse chain has operator norm at most one at every input. Equality holds whenever the stage-wise residual subspaces have a nontrivial common orthogonal complement. Let

$$\mathcal{N}_{j:K} = \bigcap_{k=j+1}^{K} \ker(W_k). \tag{44}$$

Then

$$\dim \mathcal{N}_{j:K} \ge d - \sum_{k=j+1}^{K} \mathrm{rank}(W_k). \tag{45}$$

If $\mathcal{N}_{j:K}$ contains a nonzero vector, every stage's forward and inverse Jacobian acts as the identity on that vector, and the chain's operator norm equals one. The condition depends on the learned subspaces; Equation (45) gives a sufficient criterion in terms of their ranks.

A.6 Relation to Convex-Gradient and Monotone Networks

A Monotone Gradient Network with one hidden layer includes maps of the form

$$\mathcal{G}(x) = Ax + c + W^\top \phi(Wx + b), \qquad A \succeq 0, \tag{46}$$

where $A \in \mathbb{R}^{d\times d}$, $c \in \mathbb{R}^d$, $W \in \mathbb{R}^{r\times d}$, $b \in \mathbb{R}^r$, and ϕ is elementwise monotone (Chaudhari et al., 2023; 2025). Setting $A = I$, $c = t_k$, $W = W_k$, $b = b_k$, and $\phi_\ell(s) = \tau_k a_{k,\ell}\sigma(s)$ gives Equation (8) under WTN's direction, weight, and scale constraints.

A.7 Optimal-Transport Interpretation

The WTN potential is 1-strongly convex, so $T_k = \nabla\psi_k$ is cyclically monotone. Let $\mathcal{P}_2(\mathbb{R}^d)$ denote probability measures with finite second moments. For an absolutely continuous $\mu \in \mathcal{P}_2(\mathbb{R}^d)$, let $\nu = T_{k\#}\mu$ denote its pushforward under T_k. The Lipschitz bound in Equation (35) implies $\nu \in \mathcal{P}_2(\mathbb{R}^d)$. Brenier's theorem identifies T_k as the unique μ-almost-everywhere optimal transport from μ to ν for the quadratic cost (Brenier, 1991; Rockafellar, 1970; Peyré & Cuturi, 2019).

B Classification and Historical Retrieval

B.1 Datasets and Comparison Methods

Table 4 gives the class hierarchies and split sizes for the four datasets introduced in Section 4.1.

Table 4: Datasets, category hierarchies, number of migrations, and split sizes.

Dataset	Number of classes by hierarchy level	Migrations	Train / val / test
iNaturalist 2019	$3 \to 4 \to 9 \to 34 \to 57 \to 72 \to 1{,}010$	6	238,691 / 26,522 / 3,030
FungiTastic	$3 \to 7 \to 27 \to 93 \to 307 \to 894 \to 2{,}756$	6	430,127 / 88,844 / 91,058
Shopify	$14 \to 27 \to 57 \to 131 \to 292$	4	5,628 / 1,407 / 1,671
BIOSCAN-5M	$9 \to 42 \to 606 \to 1{,}147 \to 4{,}930 \to 11{,}846$	5	289,203 / 14,757 / 39,373

All methods share c_0 and follow the information boundary in Section 3.1. Subsequent stages read each method's own preceding cache and the current-level labels.

Quantization. At each generation, we use an eight-level uniform quantizer independently for each coordinate. Its endpoints are the 0.001 and 0.999 quantiles of transformed training features. We clip values to this interval and round them to the nearest level before decoding. The quantizer is recalibrated for the selected checkpoint and frozen before test evaluation.

WTN rank and scale. Ranks are selected from $\{8, 16, 32, 64\}$ by final-generation quantized validation accuracy (Table 5). Fixed-scale WTN uses $\tau_k = 3.8$. Learnable-scale WTN starts at $\tau_k = 1.8$, with a fixed upper bound of 3.8 on iNaturalist, FungiTastic, and BIOSCAN-5M and 2.0 on Shopify.

Table 5: WTN ranks selected on validation data for the hierarchical benchmark.

Dataset	Fixed scale	Learnable scale
iNaturalist 2019	64	16
Shopify	8	8
FungiTastic	16	32
BIOSCAN-5M	32	32

Comparison methods. Table 7 lists all 15 configurations. The following controls share the generation-0 cache, data splits, hierarchy order, classifier architecture, training epochs, and quantization budget.

For CPL, we use a contractive residual branch with Lipschitz bound 0.9 and hidden dimension 16, and invert it by fixed-point iteration.

Repeated Raw-Q applies quantization without a representation transform. The parameter-matched nonlinear low-rank residual has no historical inverse; its linear counterpart removes the input-dependent gating. Signed-WTN (fixed τ) allows negative direction coefficients under the fixed-scale protocol. Free-signed WTN learns signed coefficients and normalizes them by their ℓ_1 norm. The low-rank controls compare alternative parameterizations under the same parameter budget.

Keep-All trains an adapter at each generation and retains its 3-bit cache. HBCT trains with historical heads and serves their predictions. Table 6 distinguishes representation recovery, prediction through known heads, and direct cache access. Cache counts exclude the shared model and quantizer state.

Table 6: Historical service interfaces and sample-scaled cache storage.

Method	Historical service path	Caches	Output exposed to consumer	B/vector
WTN	Structural inverse chain	One	Recovered requested-version representation	144
iResNet	Contractive-residual inverse chain	One	Recovered requested-version representation	144
BiLipNet	Operator-splitting inverse chain	One	Recovered requested-version representation	144
CP-Flow	Convex-optimization inverse chain	One	Recovered requested-version representation	144
CPL	Fixed-point inverse chain	One	Recovered requested-version representation	144
BLNN	Dual inverse chain	One	Recovered requested-version representation	144
BC-Aligner	Learned backward maps	One	Approximate historical representation	144
HBCT	Invoke known historical head	One	Historical prediction only	144
Keep-All	Read retained historical cache	5-7	Retained requested-version representation	720-1008
Repeated Raw-Q	Serve shared coordinates directly	One	Shared-coordinate representation	144

Statistical analysis. Task results are reported as means and sample standard deviations over five independent training runs. Paired comparisons match corresponding training runs within each dataset. For the main benchmark, Holm correction is applied separately within each dataset and endpoint, covering 26 accuracy comparisons and 22 historical-degradation comparisons per dataset.

B.2 Complete Task Results

Table 7 reports all configurations, including both WTN scales and the signed and low-rank controls.

Table 7: Results for all 15 configurations after the final migration. Each cell reports latest-task accuracy (%, ↑) / maximum cumulative historical-task degradation (percentage points, ↓), as mean ± sample standard deviation. Yellow cells identify the WTN scale selected from validation data; boldface marks the highest mean latest-task accuracy in each dataset.

Method	iNaturalist 2019	Shopify	FungiTastic	BIOSCAN-5M
Repeated Raw-Q	63.452±0.710 0.000±0.000	39.497±0.778 0.000±0.000	50.906±0.309 0.000±0.000	44.918±0.103 0.000±0.000
WTN (fixed scale)	63.386±0.436 0.574±0.331	39.808±0.702 0.323±0.201	50.981±0.293 1.395±0.362	45.383±0.146 1.028±0.322
WTN (learnable scale)	**63.578**±0.488 0.528±0.249	39.557±0.885 0.144±0.150	**51.207**±0.266 1.027±0.130	**45.407**±0.137 0.881±0.482
iResNet	62.660±0.501 2.079±0.772	**39.952**±0.748 3.052±0.745	50.447±0.191 2.434±0.477	43.495±0.267 5.193±2.227
BiLipNet	58.033±0.576 2.297±0.511	37.163±0.747 5.458±2.917	46.438±0.132 48.306±6.223	40.729±0.350 48.228±8.068
CP-Flow	59.109±0.386 1.142±0.771	39.856±0.635 0.287±0.375	44.502±0.614 4.645±2.908	42.670±0.507 5.261±3.273
CPL	58.944±0.798 30.330±5.281	38.049±0.810 5.087±2.243	44.812±0.234 59.090±7.390	38.376±0.439 12.697±2.534
BLNN	63.195±0.515 0.825±0.170	39.665±0.883 0.395±0.353	50.875±0.323 10.230±2.279	44.502±0.229 1.192±0.415
BC-Aligner	55.531±1.092 3.875±2.437	27.528±0.236 1.388±0.716	46.527±0.141 15.194±6.165	38.886±0.503 13.292±4.580
HBCT	62.719±1.672 0.046±0.068	39.366±0.817 0.838±0.579	50.742±0.537 0.476±0.600	44.868±0.099 0.043±0.025
Keep-All	62.139±0.540 0.000±0.000	38.372±1.042 0.000±0.000	49.456±0.186 0.000±0.000	43.547±0.098 0.000±0.000
Signed-WTN (fixed τ)	62.785±0.828 1.545±0.458	39.617±0.638 0.347±0.347	50.718±0.273 9.236±1.601	44.022±0.458 3.446±0.966
Free-signed WTN	62.990±0.683 1.195±0.358	39.593±0.629 0.347±0.347	50.532±0.259 8.578±1.724	44.367±0.403 7.990±1.869
Linear low-rank residual	45.281±2.668 *n/a*	36.709±0.584 *n/a*	35.937±1.423 *n/a*	36.299±0.380 *n/a*
Low-rank residual	56.838±1.119 *n/a*	36.517±1.203 *n/a*	44.653±0.431 *n/a*	38.984±0.451 *n/a*

The linear and nonlinear low-rank controls have no historical inverse path; their historical-degradation entries are not applicable.

WTN's latest-task advantage over independently adapted Keep-All survives Holm correction on iNaturalist, FungiTastic, and BIOSCAN-5M, but not Shopify (adjusted $p = 0.0356$, 0.000161, 0.000699, and 0.3518, respectively). This comparison includes differences in adaptation as well as retention; the storage oracle in Section 4.2 isolates retention alone.

HBCT has lower historical degradation on iNaturalist, FungiTastic, and BIOSCAN-5M when serving the heads included in training. Repeated Raw-Q has zero recovery RMSE; historical accuracy and predictions remain unchanged to numerical precision.

Fixed and learnable WTN scales are compared across four datasets and two task metrics, with Holm correction over the resulting eight tests. Only the FungiTastic latest-accuracy difference survives correction, favoring the learnable scale by 0.2256 percentage points (adjusted $p = 0.0196$). The other seven differences are not significant.

B.3 Historical Exact-Retrieval Evaluation

At historical generation j, original test queries c_j^{test} and queries $\widehat{c}_j^{(K),\text{test}}$ recovered by the frozen finite-step inverse chain search the same gallery c_j^{val}. We use the decoded validation cache deployed at generation j as the fixed gallery. Exact top-10 overlap compares neighbor-ID sets, while cosine-score RMSE compares similarity vectors over the full gallery.

We evaluate retrieval on 1,024 test queries using models fixed beforehand by validation-based selection.

Table 8: Retrieval results for validation-selected WTN. Entries are mean ± sample SD. Higher exact top-10 overlap and lower cosine-score RMSE indicate better recovery.

Dataset	Distance	Exact Top-10 overlap	Cosine-score RMSE
iNaturalist 2019	1	0.986074 ± 0.001007	0.006094 ± 0.001784
iNaturalist 2019	2	0.969492 ± 0.008406	0.016008 ± 0.001552
iNaturalist 2019	3	0.955313 ± 0.016139	0.028430 ± 0.002348
iNaturalist 2019	4	0.940723 ± 0.022437	0.044051 ± 0.008110
iNaturalist 2019	5	0.932246 ± 0.023158	0.058219 ± 0.009891
iNaturalist 2019	6	0.912246 ± 0.025367	0.089895 ± 0.019777
Shopify	1	0.983770 ± 0.003892	0.011649 ± 0.001113
Shopify	2	0.976055 ± 0.005753	0.023083 ± 0.002822
Shopify	3	0.971699 ± 0.006522	0.035307 ± 0.003598
Shopify	4	0.967461 ± 0.005585	0.052621 ± 0.005687
FungiTastic	1	0.980801 ± 0.002630	0.005087 ± 0.002305
FungiTastic	2	0.969160 ± 0.003394	0.009953 ± 0.003354
FungiTastic	3	0.953906 ± 0.003335	0.017121 ± 0.003631
FungiTastic	4	0.936230 ± 0.005207	0.038395 ± 0.004164
FungiTastic	5	0.910195 ± 0.006531	0.066300 ± 0.002769
FungiTastic	6	0.900645 ± 0.007869	0.081333 ± 0.003816
BIOSCAN-5M	1	0.988965 ± 0.001319	0.006563 ± 0.000615
BIOSCAN-5M	2	0.981563 ± 0.002518	0.012672 ± 0.000985
BIOSCAN-5M	3	0.972754 ± 0.004022	0.020522 ± 0.001404
BIOSCAN-5M	4	0.963477 ± 0.004076	0.030102 ± 0.001118
BIOSCAN-5M	5	0.949766 ± 0.005288	0.046555 ± 0.002792

Mean overlap decreases with version distance (Table 8). Table 9 compares WTN with iResNet, BiLipNet, CP-Flow, CPL, and BLNN at the oldest historical version using individual frozen checkpoints.

Table 9: Exact top-10 overlap at the oldest historical version for WTN and the structural-inverse baselines, using frozen checkpoints.

Method	iNaturalist 2019	Shopify	FungiTastic	BIOSCAN-5M
WTN (learnable scale)	0.9179	0.9675	0.9043	0.9413
WTN (fixed scale)	0.9040	0.9613	0.8918	0.9306
BLNN	0.8821	0.9699	0.8417	0.8860
iResNet	0.8580	0.9234	0.8710	0.7761
CP-Flow	0.8081	0.9656	0.6762	0.8622
BiLipNet	0.7271	0.8479	0.5306	0.7600
CPL	0.3150	0.6831	0.0303	0.2866

C Structural Constraints and Recovery Error

Single-factor ablations remove one constraint from full WTN, whereas the constraint ladder adds constraints successively to a more general convex-gradient map. Inverse-chain probes and the CPL reconstruction test distinguish perturbation amplification from numerical inversion error.

C.1 Single-Factor Ablations and Fixed Coefficient Signs

Figure 6 evaluates five one-at-a-time relaxations of

$$T(x) = \alpha x + t + \tau \sum_{\ell=1}^{r} a_\ell \sigma(w_\ell^\top x + b_\ell) w_\ell.$$

The full WTN reference fixes $\alpha = 1$ and uses the constraints in Section 3.2. Each ablation changes only the parameterization in Table 10.

Table 10: Single-factor arm definitions. Raw trainable scalars are denoted by u or v_ℓ.

Variant	Change from full WTN
Learnable α	$\alpha = 0.25 + \mathrm{softplus}(u)$, initialized at one; the skip is retained.
Learnable signs	$a_\ell = \tanh(v_\ell)/\max\{\sum_i \lvert\tanh(v_i)\rvert, \varepsilon_{\mathrm{coef}}\}$, where $\varepsilon_{\mathrm{coef}}$ is machine epsilon; initialized at $a_\ell = 1/r$ and allowed to cross zero.
Unnormalized W	Learn raw directions without row normalization, starting from the same effective unit vectors.
Unnormalized a	$a_\ell = \mathrm{softplus}(v_\ell)$, initialized at $1/r$; positivity is retained but the sum-to-one constraint is removed.
Unbounded τ	$\tau = \mathrm{softplus}(u)$; remove only its upper bound.

Signed-coefficient initialization. Both Signed-WTN configurations use fixed alternating coefficient signs and learned magnitudes, differing in whether the residual scale is fixed or learnable. The Learnable signs variant instead starts with positive coefficients and allows their signs to change during training. Its initial coefficients are uniform, and its total absolute coefficient mass is controlled. Across 105 stage checkpoints, its largest negative mass is 0.069%, 0%, 0.771%, and 0.032% on iNaturalist, Shopify, FungiTastic, and BIOSCAN-5M. Signed-WTN (fixed τ) in Table 7 starts with 50% negative mass and uses a different inverse budget.

Signed-WTN (learnable τ) in Figure 4 uses $a_\ell = s_\ell \,\mathrm{softmax}(v)_\ell$, where $s_\ell \in \{-1, 1\}$ follows the fixed alternating pattern. Initially, half of the absolute coefficient mass is negative. It shares full WTN's bounded learnable scale and other initial parameters; the imposed signs change the initial effective map.

Matched comparisons. Full WTN and its five ablations share initialization, training conditions, and quantization; each variant then migrates its own cache. On Shopify, the full WTN reference uses a learnable scale, whereas the main benchmark selects the fixed scale.

Recovery, probes, and statistics. Recovery and gain measurements use double-precision damped Newton reference inverses; the main latency measurements use the fixed-point deployment solver.

For each model, 512 deterministic test probes measure finite-difference gain from the final cache to generation 0 with perturbation norm 10^{-3}. Figure 6(a) summarizes the maximum finite-difference gain over these 512 probes for each model. Panels (b,c) summarize paired differences from full WTN using means and sample SDs. Learning the base coefficient α raises the mean sampled maximum gain above one on all four datasets.

Paired t tests compare independently trained models, with Holm correction across 40 comparisons: four datasets, five ablations, and two task endpoints.

C.2 Multi-Generation Constraint Ladder

The MGN-style ladder successively replaces a learned positive base by the identity, normalizes directions, imposes simplex weights, and bounds the residual scale. Variants with an identity base and positive coefficients already have non-expansive exact inverses. The remaining constraints control the residual Jacobian and fixed-point contraction.

Historical endpoints. Final-cache degradation follows Equation (6), maximizing over $j < K$. Full-trajectory degradation also maximizes over migration time, considering all $j < k \le K$.

On FungiTastic, complete WTN improves latest accuracy over the simplex-weighted variant (Holm-adjusted $p = 0.0129$). Its lower mean historical degradation over the full trajectory is not significant after correction ($p = 0.3034$). Paired t tests use separate Holm corrections for 48 adjacent-variant comparisons and 48 comparisons against full WTN, across four datasets and three endpoints. These families overlap. Table 11 gives both historical endpoints.

Numerical accuracy and cost. Table 12 reports costs for Newton-based inversion, separately from the main benchmark's fixed-point timings in Table 15.

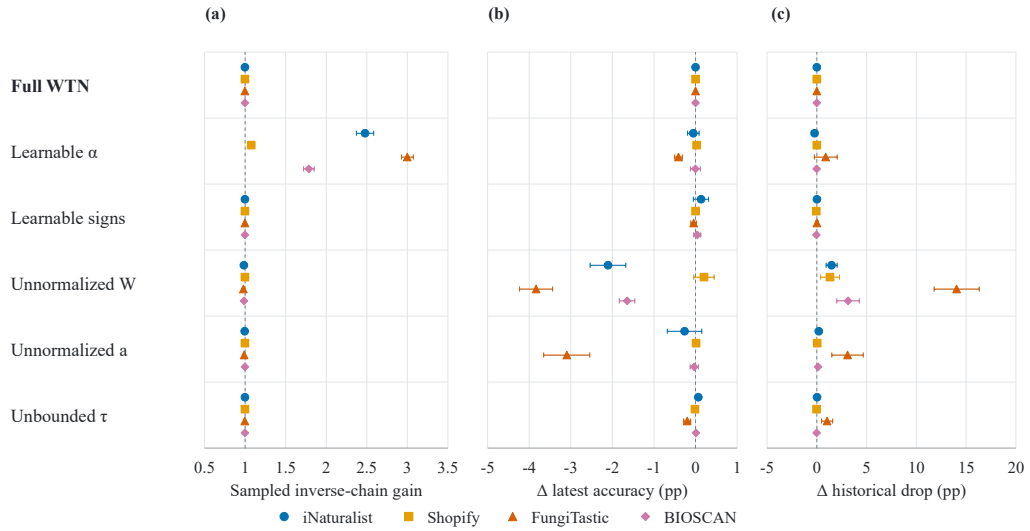


Figure 6: Matched single-factor ablations across four datasets. (a) Maximum sampled inverse-chain gain from the final cache to generation 0. (b) Latest-task accuracy and (c) maximum historical degradation, expressed as paired differences from full WTN in percentage points (ablation minus full WTN). Points and bars show means and sample standard deviations. The Learnable signs variant starts with positive coefficients and can learn their signs, unlike the fixed signs in Figure 4.

Table 11: Constraint-ladder results (mean ± sample SD). Accuracy is in %; historical drops are in percentage points. Full trajectory includes all migration times; final cache includes only targets recovered after the last migration.

Structure	Latest accuracy	History drop Full trajectory	History drop Final cache
iNaturalist			
Learned base	56.686±1.339	2.040±0.430	2.040±0.430
Identity base	57.492±0.440	2.812±0.488	2.812±0.488
Normalized directions	62.805±0.596	0.997±0.253	0.990±0.255
Simplex weights	63.432±0.445	0.772±0.283	0.706±0.296
Complete WTN	63.386±0.449	0.640±0.347	0.614±0.327
Shopify			
Learned base	39.785±0.802	1.628±0.937	1.628±0.937
Identity base	40.024±0.845	1.807±1.578	1.807±1.578
Normalized directions	39.844±0.818	0.335±0.172	0.323±0.156
Simplex weights	39.713±0.711	0.311±0.252	0.299±0.236
Complete WTN	39.808±0.702	0.359±0.203	0.323±0.201
FungiTastic			
Learned base	40.274±0.703	6.782±3.914	6.782±3.914
Identity base	41.632±0.772	8.187±3.034	8.187±3.034
Normalized directions	44.631±0.537	7.471±2.996	7.471±2.996
Simplex weights	49.466±0.223	6.632±2.602	6.542±2.511
Complete WTN	51.035±0.290	1.433±0.355	1.389±0.362
BIOSCAN-5M			
Learned base	42.196±0.499	7.520±2.528	7.520±2.528
Identity base	42.518±0.184	8.224±1.952	8.224±1.952
Normalized directions	45.296±0.078	1.326±0.451	1.326±0.451
Simplex weights	45.371±0.129	1.071±0.262	1.071±0.262
Complete WTN	45.365±0.150	1.054±0.327	1.041±0.323

Table 12: Inverse-chain P50 latency (ms) at batch size one, using float32 on an exclusive L20. Task heads, unpacking, and transfer are excluded. Budget is per stage; damped Newton may stop early.

Dataset	Structure	Solver	Budget	P50 (ms)
iNaturalist	Learned base	Damped Newton	100	74.173
	Identity base	Damped Newton	100	77.354
	Normalized directions	Newton	3	10.175
	Simplex weights	Newton	3	10.186
	Complete WTN	Newton	3	10.250
Shopify	Learned base	Newton	3	6.737
	Identity base	Newton	3	6.593
	Normalized directions	Newton	2	4.821
	Simplex weights	Newton	2	4.845
	Complete WTN	Newton	2	4.896
FungiTastic	Learned base	Damped Newton	100	78.242
	Identity base	Damped Newton	100	84.824
	Normalized directions	Damped Newton	100	53.800
	Simplex weights	Newton	5	15.677
	Complete WTN	Newton	3	9.974
BIOSCAN-5M	Learned base	Damped Newton	100	47.287
	Identity base	Damped Newton	100	45.488
	Normalized directions	Newton	3	8.342
	Simplex weights	Newton	2	5.941
	Complete WTN	Newton	2	6.092

C.3 Sampled Inverse-Chain Gains

For the frozen checkpoints in Table 2, analytic inverse-chain gains agree with finite differences along the same 512 sampled directions. For WTN, any nonzero direction in the common nullspace is unchanged by every inverse Jacobian, making the global chain norm equal to one even when all sampled gains are below one (Appendix A.5).

C.4 CPL Numerical Reconstruction and Historical Recovery

The CPL reconstruction check in Section 4.3 uses a clean reconstruction RMSE threshold of 10^{-5}. The frozen model satisfies this threshold with 200 inverse iterations per stage.

The six-stage inverse takes 937.278 ms at P50 for batch size one on an exclusive NVIDIA L20. Table 15 reports the main benchmark's 100-iteration setting.

D Storage and Computational Cost

D.1 Parameter and Operation Counts

A fixed-scale WTN block with embedding dimension d and r learned directions contains

$$rd + 2r + d \tag{47}$$

trainable parameters: rd for the raw direction matrix, r for direction biases, r for coefficient logits, and d for the translation. The learnable-scale setting adds one scalar γ. At $d = 384$ and $r = 16$, the two settings contain 6,560 and 6,561 parameters, respectively.

The projection and back-projection in Equation (24) cost about $2Brd$ multiply-accumulate operations per batch, plus $O(Br)$ for gating. An M-step inverse costs about $2MBrd$ operations; a chain from generation K to j costs about $2(K - j)MBrd$ when stages share r and M.

D.2 Cache Payload and Serving Tensor State

Cache payload is Ndb bits for WTN and $(K+1)Ndb$ bits for Keep-All, or $db/8$ and $(K+1)db/8$ bytes per vector. At $d = 384$ and $b = 3$, one cache uses 144 bytes per vector. Adapters, heads, and quantizer states are retained separately and do not scale with the number of samples N.

Table 13 adds adapters, heads, and quantizer tensors to the ideal cache payload. Keep-All here uses independently trained adapters, as in Table 7.

Table 13: Calculated storage of the ideal cache payload and retained serving tensors (bytes). Ratios are Keep-All/WTN.

Dataset	Selected WTN	Keep-All	Ratio
iNaturalist	40,640,092	272,419,876	6.70
Shopify	2,179,396	7,205,956	3.31
FungiTastic	94,320,196	621,403,084	6.59
BIOSCAN-5M	78,205,876	325,420,496	4.16

D.3 Service and Materialization Latency

Table 15 reports P50 latency for serving the oldest historical version at batch size one on an exclusively allocated NVIDIA L20. Inputs are decoded float32 tensors resident on the GPU, so unpacking and host-to-device transfer are excluded.

Inverse iteration counts. With model parameters and quantizers fixed, we test $m \in \{1, 2, 3, 5, 8, 10, 20, 50, 100\}$ iterations on at most 256 validation samples. The same validation criteria apply to all structural-inverse methods, using each method's own 100-iteration output as the reference. Clean reconstruction RMSE measures recovery of an input after a forward pass without quantization; cached RMSE compares the candidate inverse with the reference on decoded caches. A candidate must satisfy

$$\begin{aligned} \mathrm{RMSE}_{\mathrm{clean}}(m) &\le \max\left\{10^{-5},\ 1.05\ \mathrm{RMSE}_{\mathrm{clean}}(100) + 10^{-8}\right\}, \\ \mathrm{RMSE}_{\mathrm{cached}}(m, 100) &\le 10^{-5}, \end{aligned} \tag{48}$$

and differ from the reference in at most one prediction by the preceding task head. We select the smallest qualifying count, or 100 if none qualifies, and fix it before test evaluation. WTN, iResNet, BiLipNet, and CPL execute the selected iteration count without early stopping; CP-Flow and BLNN use it as an iteration limit. CP-Flow uses L-BFGS with gradient and change tolerances of 10^{-9}. BLNN stops when the mean root-residual norm is at most 10^{-6} or backtracking yields no further reduction. Table 14 gives the budgets supplied to each method's solver for the timed checkpoints.

Table 14: Per-stage inverse iteration budgets for the checkpoints used in the latency measurements. Sequences follow the forward stage order $1, \ldots, K$; a single value applies to every stage.

Method	iNaturalist 2019	Shopify	FungiTastic	BIOSCAN-5M
WTN (fixed scale)	20, 8, 8, 8, 8, 8	8, 8, 5, 8	8, 8, 8, 5, 8, 10	8, 10, 10, 5, 8
WTN (learnable scale)	10, 8, 5, 5, 5, 5	8, 5, 5, 5	8, 8, 8, 5, 8, 8	8, 8, 8, 5, 5
iResNet	8, 3, 5, 5, 8, 20	3, 10, 8, 5	5, 5, 8, 5, 5, 20	8, 5, 8, 8, 8
BiLipNet	20	20	20	20
BLNN	20, 8, 10, 10, 10, 8	3, 5, 8, 3	10, 8, 20, 10, 10, 10	8, 20, 8, 8, 8
CP-Flow	20, 20, 20, 8, 5, 8	10, 8, 8, 8	50, 20, 20, 8, 8, 8	20, 8, 8, 8, 8
CPL	100	100	100	100

iResNet is the fastest structural-inverse method; WTN is faster than BiLipNet, BLNN, CP-Flow, and CPL.

Latest-task service requires no inverse and takes 0.047-0.065 ms at batch size one across methods. Offline materialization of all four to six WTN migrations costs 0.985-1.568 ms per vector at the same batch size.

Table 15: P50 model-side latency (ms) for serving the oldest version at batch size one from decoded GPU-resident inputs. Boldface marks the fastest structural-inverse method.

Method	iNaturalist 2019	Shopify	FungiTastic	BIOSCAN-5M
Repeated Raw-Q	0.048	0.054	0.047	0.047
WTN (chosen scale)	4.942	3.198	6.206	4.373
iResNet	**4.194**	**2.285**	**3.967**	**3.089**
BiLipNet	23.967	17.586	24.934	19.829
BLNN	95.891	30.858	73.779	56.819
CP-Flow	189.634	88.996	169.199	163.214
CPL	458.959	317.179	464.543	397.172
BC-Aligner	0.219	0.163	0.208	0.184
HBCT	0.049	0.048	0.048	0.048
Keep-All	0.056	0.049	0.047	0.048

Historical interfaces are defined in Table 6. CPL uses 100 iterations per stage; its 200-iteration FungiTastic result is in Appendix C.4.

E Application and Encoder-Upgrade Experiments

E.1 DISC21 Retrieval and Legacy Verification

Evaluation scope. The DISC21 evaluation (Douze et al., 2021) uses the first 50,000 official references, all 529 development queries with an annotated match in that subset, and 529 queries with no match in the complete development ground truth. These queries are held out from training and selection of hyperparameters and thresholds.

Generation 0 directly quantizes unit-normalized DINOv2-S CLS features, without the hierarchical benchmark's supervised stem. The two migrations adapt the decoded caches to photometric and compression perturbations, then geometric and partial-content perturbations, using contrastive training.

Service and metrics. The legacy cosine threshold is selected on validation data and held fixed after migration. A query is encoded once and passed through the codec and adapter chain, then retrieves 100 candidates by exact cosine search in the latest gallery. Search uses normalized vectors. Recovery applies both WTN inverse stages to the unnormalized decoded query and candidate vectors. The service returns the candidate with highest recovered generation-0 cosine score if it exceeds the fixed threshold, and rejects the query otherwise. Only the latest packed gallery is retained; old caches are used for offline comparison.

Recall@10 and Recall@100 count candidate hits among all 529 positive queries. Micro-AP uses the official code on the returned candidate scores. Service recall counts correct returns among all positives; conditional acceptance counts accepted true matches among positives retrieved in the Top-100. Retention is measured on the original service's 60 correct returns. Table 16 reports these metrics. Candidate-pair agreement is $99.9994 \pm 0.0005\%$, dominated by joint rejections, whereas query recovery RMSE is $(2.043 \pm 0.036) \times 10^{-4}$.

Table 16: DISC21 candidate retrieval and verification on the 50,000-reference subset. Percentages use the denominators defined in the text. The original cache and verifier are fixed; WTN values are mean $\pm$ sample SD.

Metric	Original service	WTN inverse service
Candidate Recall@10 (%)	64.839	65.671 ± 0.104
Candidate Recall@100 (%)	71.834	73.422 ± 0.104
Candidate micro-AP (%)	35.053	34.408 ± 0.069
Service recall (%)	11.342	11.229 ± 0.104
Conditional true-match acceptance (%)	15.789	15.294 ± 0.139
Originally correct queries retained (%)	100.000	99.000 ± 0.913
Incorrect returns on 529 negative queries	0	0

Runtime storage. The packed gallery uses 7.2 MB. Startup creates two 76.8 MB FP32 matrices: unnormalized vectors for recovery and normalized vectors for search. Models, identifiers, and temporary tensors add further memory.

Service latency. On an exclusively allocated NVIDIA L20, repeated calls for one fixed query and 100 candidates give feature-to-decision latencies of 79.745 ms at P50 and 80.452 ms at P95. Including local image decoding, preprocessing, encoding, and query transfer gives 99.715 and 100.391 ms, respectively. Gallery initialization takes 86.804 ms at P50 and is timed separately. Timing uses individual query encoding rather than the batch encoding used for evaluation; some retrieved candidates differ, but both modes reject this query.

Case selection. Using one fixed model, we select the correctly returned positive closest to the median verifier margin for Figure 5(a). For panel (b), we select the positive with highest query recovery RMSE among those whose true reference is retrieved but not correctly returned. These rules are applied before viewing images, with query IDs breaking ties. The selected queries are Q07482 and Q19485; the latter is also rejected by the original verifier.

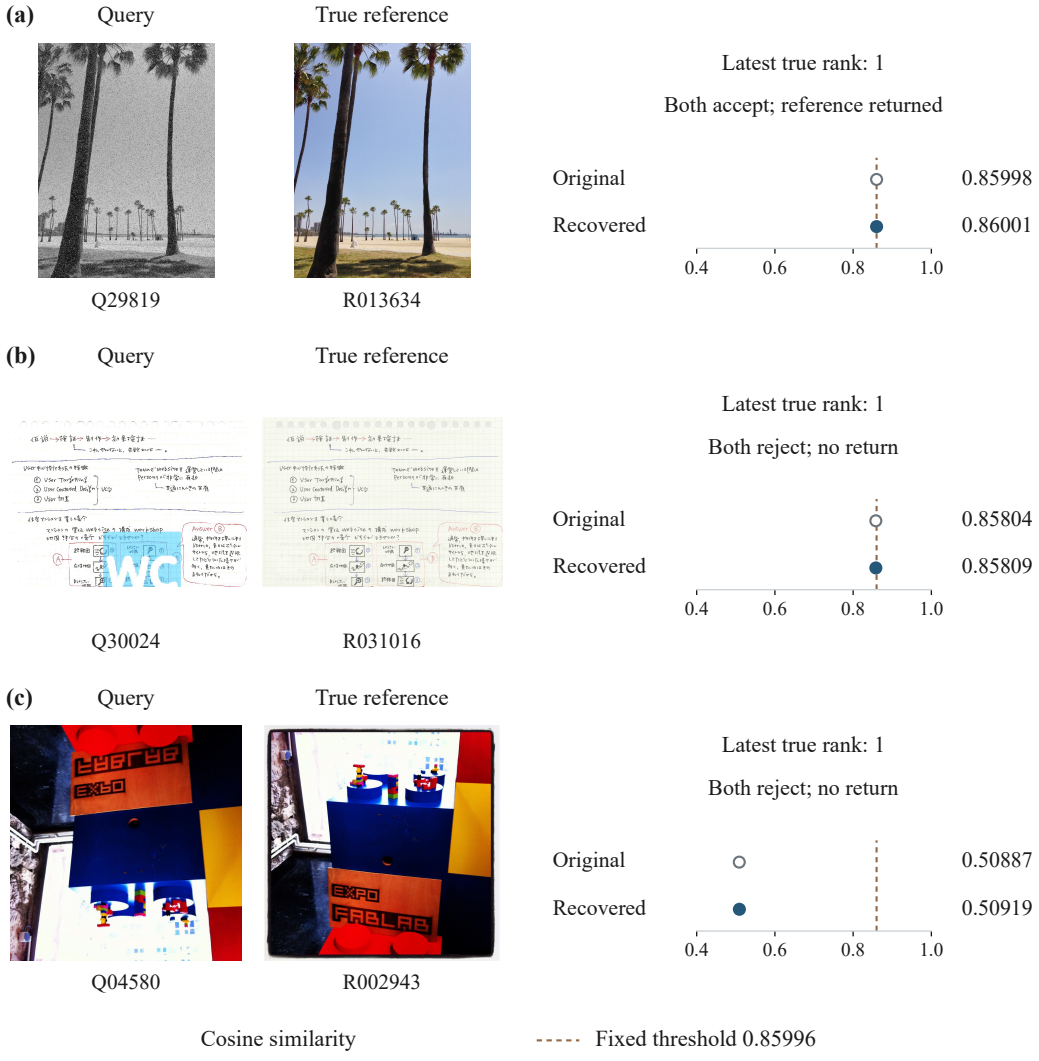


Figure 7: DISC21 examples with a fixed legacy verifier. Each true reference ranks first in the latest cache. Both original and recovered scores exceed the threshold in (a) and fall below it in (b,c). Original and recovered scores are defined as in Figure 5. Values are displayed to five decimal places; decisions use full precision.

For Figure 7, we select queries distinct from those in the main figure whose true references are retrieved in the Top-100. Using recovered scores after evaluation, we choose the accepted and rejected pairs closest to the threshold for (a,b), and a rejected pair near the lower quartile for (c).

E.2 Two Successive Encoder Upgrades

1024-dimensional test evaluation. Table 3 uses bridges fitted on all training pairs. The first 384 coordinates are the frozen Procrustes projection; the remaining coordinates retain the orthogonal component of the new encoder's features. DINOv2-B has 256 trailing zero coordinates, and the generation-0 cache has 640. All methods consume the preceding 3-bit cache at each upgrade. Latest-task accuracy uses all test images; mixed-gallery mAP@10 uses 1,024 test queries against the full validation gallery, alternating migrated and new-encoder entries. The generation-0 classifier is shared across methods within each training run. Learned baselines use fixed architectures and 30 training epochs. Inversion uses 100 steps.

For WTN, validation selected ranks $(32, 32)$ and scales $(3.5, 3.5)$ on Aircraft, and ranks $(16, 1024)$ and scales $(0.1, 1.0)$ on iNaturalist-Loc500, for the first and second upgrades respectively. The search covered ranks, scales, learning rates, and codecs, with up to 60 training epochs. We also evaluate WTN with $r = 16$ and $\tau = 3.8$ at both stages. Both WTN configurations were frozen before test evaluation.

All WTN metrics in Tables 3 and 18 come from a single validation-selected configuration per dataset. Table 17 reports both WTN configurations in full.

Table 17: Test results for both WTN configurations at 1024 dimensions (mean ± sample SD). Historical drop is measured in percentage points.

Dataset	Configuration	Accuracy (%)	mAP@10	Max. hist. drop
Aircraft	Validation-selected	65.065 ± 0.383	0.1481 ± 0.0011	0.144 ± 0.164
	Fixed $r = 16$	61.722 ± 0.633	0.1501 ± 0.0005	0.246 ± 0.193
iNaturalist-Loc500	Validation-selected	76.819 ± 0.146	0.3707 ± 0.0004	0.011 ± 0.031
	Fixed $r = 16$	76.059 ± 0.166	0.3842 ± 0.0007	1.525 ± 0.865

Table 18: Maximum historical-task accuracy drop after two encoder upgrades at 1024 dimensions (percentage points; mean ± sample SD). The nonlinear low-rank adapter has no inverse.

Method	Aircraft	iNaturalist-Loc500
No migration	0.000 ± 0.000	0.000 ± 0.000
Procrustes	1.206 ± 0.140	0.054 ± 0.033
Nonlinear low-rank	n/a	n/a
iResNet	0.276 ± 0.133	2.048 ± 0.105
WTN	0.144 ± 0.164	0.011 ± 0.031

384-dimensional anchor study. We evaluate DINOv2-S → DINOv2-B → DINOv2-L on Aircraft and iNaturalist-Loc500 at anchor fractions of 1%, 5%, 10%, and 100%. Lossy rectangular Procrustes bridges map native features of dimensions 384, 768, and 1024 into a shared 384-dimensional space. Anchors determine bridge fitting and paired alignment; current-task cross entropy uses new-encoder features from the entire training set.

Mixed galleries and historical evaluation. Methods share bridges and nested anchor subsets within each dataset and anchor fraction. Mixed galleries combine migrated old entries with new-encoder entries under the method's codec; queries use the new encoder and the same shared space. The no-migration control reuses its input codec. After migration model selection, we fit the generation-0 head using the generation-0 training and validation caches, before test evaluation.

Tables 19 and 20 report accuracy and exact retrieval. WTN has the highest mean latest accuracy at all anchor fractions on iNaturalist-Loc500; nonlinear low-rank adaptation is more accurate on Aircraft. Procrustes exceeds WTN in mean mixed-gallery mAP@10 in all eight settings. WTN's mean historical accuracy change stays within 0.3 percentage points at both generations in each setting.

After Holm correction, eight comparisons favor WTN and eleven favor a comparator. No comparison between WTN and iResNet is significant; these tests do not establish equivalence. The correction

covers 176 paired t tests across two datasets, four anchor fractions, four comparators, and six endpoints. The comparison family excludes 16 inapplicable historical-recovery tests for the nonlinear low-rank control.

Table 19: Aircraft after two encoder upgrades (mean ± sample SD). Accuracy is in %, mAP@10 on a 0-1 scale, and Δ_0, Δ_1 are historical accuracy drops (pp) from the final cache.

Anchors	Method	Latest accuracy	mAP@10	Δ_0	Δ_1
1%	No migration	52.175±0.239	0.1365±0.0000	0.000±0.000	0.000±0.000
	Procrustes	48.725±0.605	0.1374±0.0000	2.022±0.335	1.134±0.446
	Nonlinear low-rank	56.652±0.619	0.1377±0.0010	*n/a*	*n/a*
	iResNet	52.007±0.355	0.1360±0.0004	0.192±0.309	0.048±0.186
	WTN	52.625±0.245	0.1351±0.0004	0.090±0.196	0.258±0.231
5%	No migration	54.023±0.769	0.1403±0.0000	0.000±0.000	0.000±0.000
	Procrustes	49.535±0.321	0.1427±0.0000	1.920±0.290	1.050±0.237
	Nonlinear low-rank	58.092±0.742	0.1407±0.0007	*n/a*	*n/a*
	iResNet	54.281±0.651	0.1399±0.0004	0.234±0.303	0.444±0.243
	WTN	54.275±0.801	0.1400±0.0005	0.168±0.235	0.198±0.205
10%	No migration	55.050±0.286	0.1433±0.0000	0.000±0.000	0.000±0.000
	Procrustes	49.949±0.468	0.1475±0.0000	2.118±0.346	1.008±0.354
	Nonlinear low-rank	59.184±0.805	0.1454±0.0010	*n/a*	*n/a*
	iResNet	55.458±0.353	0.1431±0.0006	0.354±0.096	0.216±0.278
	WTN	55.464±0.438	0.1437±0.0005	0.162±0.151	0.186±0.166
100%	No migration	57.312±0.503	0.1471±0.0000	0.000±0.000	0.000±0.000
	Procrustes	53.633±0.433	0.1506±0.0000	0.636±0.320	0.270±0.200
	Nonlinear low-rank	60.642±0.058	0.1466±0.0010	*n/a*	*n/a*
	iResNet	57.348±0.433	0.1480±0.0005	0.264±0.207	0.150±0.199
	WTN	57.798±0.628	0.1482±0.0005	0.072±0.181	0.132±0.210

Table 20: iNaturalist-Loc500 after two encoder upgrades (mean ± sample SD), with the metrics and units in Table 19. In both tables, n/a denotes the nonlinear low-rank control's unavailable inverse.

Anchors	Method	Latest accuracy	mAP@10	Δ_0	Δ_1
1%	No migration	75.598±0.175	0.3584±0.0000	0.000±0.000	0.000±0.000
	Procrustes	74.955±0.036	0.3709±0.0000	0.619±0.182	0.464±0.117
	Nonlinear low-rank	74.962±0.137	0.3574±0.0007	*n/a*	*n/a*
	iResNet	75.677±0.139	0.3603±0.0005	0.067±0.041	0.000±0.006
	WTN	75.680±0.138	0.3616±0.0008	0.037±0.075	0.000±0.038
5%	No migration	75.986±0.081	0.3645±0.0000	0.000±0.000	0.000±0.000
	Procrustes	76.034±0.118	0.3781±0.0000	0.178±0.069	0.058±0.072
	Nonlinear low-rank	75.549±0.189	0.3685±0.0011	*n/a*	*n/a*
	iResNet	76.037±0.033	0.3666±0.0009	0.090±0.093	0.093±0.059
	WTN	76.077±0.057	0.3692±0.0005	0.018±0.031	0.002±0.004
10%	No migration	76.021±0.068	0.3684±0.0000	0.000±0.000	0.000±0.000
	Procrustes	76.066±0.129	0.3808±0.0000	0.083±0.090	0.019±0.034
	Nonlinear low-rank	75.568±0.157	0.3665±0.0012	*n/a*	*n/a*
	iResNet	76.051±0.078	0.3684±0.0005	0.075±0.088	0.040±0.051
	WTN	76.173±0.140	0.3673±0.0009	0.035±0.034	0.022±0.049
100%	No migration	76.096±0.098	0.3691±0.0000	0.000±0.000	0.000±0.000
	Procrustes	76.173±0.071	0.3805±0.0000	−0.008±0.042	0.006±0.064
	Nonlinear low-rank	75.531±0.147	0.3654±0.0010	*n/a*	*n/a*
	iResNet	76.160±0.070	0.3680±0.0012	0.027±0.063	0.022±0.041
	WTN	76.203±0.123	0.3690±0.0009	−0.022±0.055	−0.003±0.007