PANORAMA: FAST-TRACK NEAREST NEIGHBORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Retrieval-augmented generation (RAG) has put nearest-neighbor search over learned embeddings on the inference path of language models via purpose-built vector indexes. An index narrows a query result to a set of candidates, each of which must be verified by its d-dimensional distance. As embeddings grow wider, this verification comes to dominate query time over index traversal. Most of this verification overhead is wasted: a distance is computed only to decide whether a candidate enters the top-k set; for most candidates the answer is "no". We introduce PANORAMA, a verification module that verifies candidates progressively. It rotates vectors into an orthogonal basis fitted to the data, accumulates distances in a piecewise manner, and abandons a candidate as soon as a Cauchy–Schwarz lower bound on its unread remainder pushes its distance past the running top-k threshold. As energy concentrates in a few principal directions, the bound becomes decisive after a small fraction of the coordinates is read. While PANORAMA's strict mode returns exactly the top-k of the supplied candidates, one parameter trades recall for throughput. A cost model ties the pruning effect to the spectral decay of the fitted basis. PANORAMA sits on top of an unmodified index: we realize the bound within inverted-list, product-quantized, graph, reranking, and RaBitQ indexes with layouts that keep the pruning loop vectorized. Across image descriptors and text embeddings, the gain grows with dimension and is largest on the wide embeddings that RAG systems search.

1 INTRODUCTION

Retrieval-augmented generation (RAG) has become the standard way to ground a language model in knowledge it was not trained on: a corpus is split into passages, each passage is embedded by a trained encoder, and the embedding of the prompt retrieves its nearest passages into the model's context (Karpukhin et al., 2020; Lewis et al., 2020; Gao et al., 2023). As RAG moves from prototypes into assistants, search products, and agents, building and serving a nearest-neighbor-search index over one's own data has become a routine part of deploying a model, in addition to recommendation and search systems that depend on such indexes (Koren et al., 2009; Pan et al., 2024); retrieval sits on the critical path of every request, so its latency and cost add to the model's own.

Meanwhile, the embeddings being searched are getting wider: classical image descriptors have around a hundred dimensions, whereas contrastively trained text encoders produce a thousand or more (Neelakantan et al., 2022; OpenAI). Approximate indexes such as inverted lists, proximity graphs, and product quantization (Jégou et al., 2008; Malkov & Yashunin, 2020; Jégou et al., 2010; Douze et al., 2025) render search sublinear by proposing a small set of candidates, but must still *verify* each proposed candidate against the running top-k set via an $O(d)$ distance calculation. As d grows, this verification, rather than the index traversal, becomes the bottleneck of retrieval.

Most of this verification work is nonetheless spent on unqualified candidates. Each distance is computed only to decide whether a candidate enters the top-k set; for most candidates the answer is "no". This decision becomes cheap in a proper coordinate basis, as the covariance spectra of learned word, sentence, and contextual embeddings decay swiftly, letting a few principal directions carry most of their energy (Mu & Viswanath, 2018; Ethayarajh, 2019; Gao et al., 2019). Such a proper basis is the principal-component basis, in which a vector's leading coordinates carry most of its norm and the remainder can be tightly bounded based on the little energy it has left.

The task of turning this observation into wall-clock speed via a verification method performing distance computations with *early abandoning* raises three challenges:

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.

- **Safety**: prior early-abandoning methods (Gao & Long, 2023; Deng et al., 2024; Yang et al., 2025) estimate the unread distance probabilistically from a sample or a learned residual model and may thus discard true neighbors; exact reranking and reproducible scorers do not accept such loss.
- **Index diversity**: inverted lists scan contiguous rows; product quantization scores codes through lookup tables; graphs visit scattered nodes one at a time; and RaBitQ scores sign codes with popcounts; an early-abandoning method must fit each layout without disturbing it.
- **Overhead**: bookkeeping per level costs branches, memory traffic, and threshold updates, while a naive loop erases the arithmetic in each iteration.

Our proposal, PANORAMA, addresses these challenges. It rotates vectors into an orthogonal basis, splits the coordinates into a few *levels*, and lower-bounds the unread distance suffix at each level by the Cauchy–Schwarz product of the query's and the candidate's remaining energies. This bound is deterministic, so in a *strict mode* PANORAMA returns the exact top-k of the supplied candidates; a parameter ε lets it trade recall for speed. Combined with the index's own knobs, this lets PANORAMA reach any recall target the index reaches. The basis poses an easy learning problem: among all orthogonal bases, PCA fitted to the data minimizes the expected unread energy at every prefix length; via a cost model, we relate the pruning effect to the spectral decay of that fit. On the representation-learning side, PANORAMA translates a statistical property of learned embeddings to a computational saving, in a training- and encoder-agnostic manner, without the nested objective of Matryoshka representations (Kusupati et al., 2022). On the systems side, level-major batches, native-layout sidecars, and a sign-prefix gate carry the same mathematical bound into contiguous, quantized, graph, reranking, and RaBitQ indexes.

Our contributions encompass the following:

- **A progressive bound with a learned basis:** we provide an exact strict mode along with an ε-relaxation thereof, prove the optimality of PCA among orthogonal bases for expected tail energy, and provide a cost model that relates pruning effect to spectral decay (Section 3).
- **Index-specific realizations:** we craft level-major batching and SIMD kernels for contiguous indexes, native-layout sidecars for scattered ones, a PQ-compatible shaping of the basis, and an IVF–RaBitQ integration whose outputs stay bit-identical (Section 4).
- **A matched-recall evaluation:** we assess PANORAMA against state-of-the-art exact and probabilistic verification methods within clustered, compressed, graph, and reranking indexes, with ablations that separate the contributions of the learned basis, the layout, and the kernel, and test whether the cost model's saved work translates to wall-clock speedup (Section 5).

2 RELATED WORK

Indexes and quantizers. Approximate nearest-neighbor search separates two decisions: which vectors to visit, and how cheaply to score each one. Inverted files (Jégou et al., 2008) and the inverted multi-index (Babenko & Lempitsky, 2015) restrict search to nearby coarse clusters, and HNSW (Malkov & Yashunin, 2020), NSG (Fu et al., 2019), and DiskANN (Subramanya et al., 2019) traverse navigable proximity graphs. Quantizers answer the second question: PQ (Jégou et al., 2010), OPQ (Ge et al., 2013), residual and additive quantization (Chen et al., 2010; Martinez et al., 2016), and anisotropic quantization (Guo et al., 2020) score compact codes through lookup tables, FastScan (André et al., 2015) keeps those tables in SIMD registers, RaBitQ (Gao & Long, 2024) scores binary and multi-bit codes with an error bound, and VAQ (Paparrizos et al., 2022) allocates bits by subspace variance. Libraries such as Faiss (Douze et al., 2025; Johnson et al., 2019) combine both choices. PANORAMA is orthogonal to the first decision and changes the second: it alters how each proposed candidate is verified, not which candidates are proposed, so it can sit on top of any existing index (on graphs, pruned nodes may still alter the traversal; see Section 4). We therefore compare full-precision and compressed-only methods separately, and treat exact reranking of a compressed shortlist as one more verification pass.

Progressive distance comparison. Rejecting a candidate with a cheap bound before computing its exact distance goes back to filter-and-refine scans such as the VA-file (Weber et al., 1998). BOND (de Vries et al., 2002) and Stepwise (Kashyap & Karras, 2011) derive deterministic early-abandoning bounds in a fixed coordinate system; ADSampling (Gao & Long, 2023) samples randomly

rotated dimensions and stops by a hypothesis test; PDX (Kuffo et al., 2025) pairs progressive comparison with a columnar layout, using either ADSampling or BOND pruning; DADE (Deng et al., 2024) calibrates a data-aware test; and DDC (Yang et al., 2025) learns the distribution of the residual distance. The same pattern powers some of the most effective early-rejection machinery in machine learning: detection cascades reject most windows with a few cheap features (Viola & Jones, 2001), and cascade ranking spends expensive scorers only on documents that survive cheap ones (Wang et al., 2011). Relative to these tests, PANORAMA contributes three things: a strict mode that is deterministic, unlike ADSampling, DADE, and DDC, and never discards a candidate in the exact top-k of the supplied set; a bound evaluated in a basis *learned* from the data, unlike those of BOND and Stepwise, which is optimal among orthogonal bases and shaped to remain usable under product quantization; further, the bound is specialized to inverted lists, PQ codes, scattered graph candidates, reranking, and RaBitQ sign codes. Since these methods differ in guarantees and layouts, we report both throughput–recall frontiers and the fraction of coordinates each reads.

Learned and reduced representations. Dimensionality reduction lowers the cost of verification before search begins: random projections preserve distances in expectation by the Johnson–Lindenstrauss lemma (Johnson & Lindenstrauss, 1984), and PCA-based reductions exploit the low effective rank of real embeddings (Wang et al., 2025). Both discard information permanently, so their recall depends on how much energy is dropped. That effective rank is low because trained embeddings are anisotropic, with a few directions dominating the variance (Mu & Viswanath, 2018; Ethayarajh, 2019; Gao et al., 2019), which is usually treated as a defect to be post-processed away. Matryoshka representation learning (Kusupati et al., 2022) instead trains embeddings whose prefixes are themselves usable, enabling coarse-to-fine retrieval with truncated vectors, and bi-metric search (Xu et al., 2024) pairs a cheap proxy distance with an exact one. PANORAMA shares the coarse-to-fine view but keeps the full vector: it learns an ordering of coordinates for any trained encoder, uses the prefix only to *bound* the full distance, and so stays exact in the strict mode. Its cost is governed by the same spectral decay that makes reduction and nested training effective, and is formalized by our cost model.

Graph routing and reranking. FINGER (Chen et al., 2023) and KS2 (Lu et al., 2025) approximate the distance tests that decide which graph neighbors to expand, and PEOs (Lu et al., 2024) precomputes per-edge routing metadata; these change the traversal, PANORAMA the verification, and the two are complementary. Because DADE ships as a fork of hnswlib with its own graph construction, we report a same-graph comparison inside Faiss and an authors-code comparison separately. Compressed-domain search is commonly followed by an exact reranking of a short list with the original vectors (Jégou et al., 2011; André et al., 2015); that refinement pass is itself a verification step, and PANORAMA accelerates it directly as well.

3 PROGRESSIVE VERIFICATION WITH A LEARNED BASIS

PANORAMA's task is to decide which of the index-proposed candidates belong to the top-k set. To achieve that, it rotates each vector into an orthogonal basis fitted to the data, splits the coordinates into L levels, and stores with each candidate vector the energy that remains after each level. At query time, it accumulates the query-candidate inner product level by level, lower-bounds the unread part of the distance at each level by the Cauchy–Schwarz inequality using the two stored energies, and discards a candidate if its lower bound exceeds the running k^{th}-best distance. As the fitted basis puts most of a vector's energy into the first few levels, the bound becomes tight early, discarding most candidates based on a small fraction of their coordinates; because the bound is true, it incurs no false dismissals of candidates in the top-k set.

The bound. Let $\mathbf{q}, \mathbf{x} \in \mathbb{R}^d$ and let T be an orthogonal transform. For cutpoints $0 = m_0 < m_1 < \cdots < m_L = d$, define the accumulated prefix inner product $p_m = \sum_{j \le m} T(\mathbf{q})_j T(\mathbf{x})_j$ and the suffix energies $R_q(m) = \sum_{j>m} T(\mathbf{q})_j^2$ and $R_x(m) = \sum_{j>m} T(\mathbf{x})_j^2$. Orthogonality preserves the distance, and the Cauchy–Schwarz inequality on the unread coordinates gives

$$\mathrm{LB}_m(\mathbf{q}, \mathbf{x}) = \|\mathbf{q}\|^2 + \|\mathbf{x}\|^2 - 2p_m - 2\sqrt{R_q(m) R_x(m)} \le \|\mathbf{q} - \mathbf{x}\|^2. \tag{1}$$

Expanding the norms, the bound equals $\sum_{j\le m}(T(\mathbf{q})_j - T(\mathbf{x})_j)^2 + \left(\sqrt{R_q(m)} - \sqrt{R_x(m)}\right)^2$: the partial distance used by earlier deterministic early-abandoning methods, plus a nonnegative correction for the unread energies, so it is never looser than the partial distance and is decisive long before it once the suffix energies are small. This bound only requires the $L+1$ suffix energies $R_x(m_\ell)$ per stored vector, tightens monotonically in m, and is exact at $m_L = d$. PANORAMA advances through the cutpoints and stops reading a candidate at the first level ℓ where LB_{m_ℓ} exceeds the running k^{th}-best exact distance d_k; Algorithm 1 gives the batched form used for contiguous indexes, in which each level corresponds to one SIMD pass over the survivors of a batch and the threshold is refreshed between batches.

Algorithm 1 PANORAMA: progressive verification of a batch-ordered candidate stream

Require: query $\mathbf{q}$; candidates C with stored $T(\mathbf{x})$, $\|\mathbf{x}\|^2$, and $R_x(m_\ell)$; cutpoints $m_0 < \cdots < m_L$; k; batch size B; $\varepsilon \in [0,1]$

1: $\tilde{\mathbf{q}} \leftarrow T(\mathbf{q})$; precompute $\|\mathbf{q}\|^2$ and $R_q(m_\ell)$ for all ℓ
2: $H \leftarrow$ exact top-k heap of the first k candidates; $d_k \leftarrow$ largest distance in H
3: **for** each batch $\mathcal{B} \subset C$ of size B, stored level-major **do**
4: $\quad S \leftarrow \mathcal{B}$; $p_{\mathbf{x}} \leftarrow 0$ for all $\mathbf{x} \in S$
5: $\quad$ **for** $\ell = 1, \ldots, L$ **do**
6: $\quad\quad p_{\mathbf{x}} \leftarrow p_{\mathbf{x}} + \sum_{j=m_{\ell-1}+1}^{m_\ell} \tilde{q}_j\, T(\mathbf{x})_j$ for all $\mathbf{x} \in S$ ▷ contiguous SIMD pass
7: $\quad\quad S \leftarrow \{\mathbf{x} \in S : \|\mathbf{q}\|^2 + \|\mathbf{x}\|^2 - 2p_{\mathbf{x}} - 2\varepsilon\sqrt{R_q(m_\ell)R_x(m_\ell)} \le d_k\}$ ▷ compact survivors
8: $\quad$ **for** each $\mathbf{x} \in S$ **do** ▷ $R(m_L) = 0$, so the distance is exact
9: $\quad\quad D \leftarrow \|\mathbf{q}\|^2 + \|\mathbf{x}\|^2 - 2p_{\mathbf{x}}$; **if** $D < d_k$ **then** push $(D, \mathbf{x})$ into H, pop its largest entry, update d_k
10: **return** H

Safety and relaxation. A candidate being pruned is always farther than the k^{th}-best candidate seen so far, and a stale threshold can only be too loose, never too tight.

Proposition 3.1 (Strict-mode safety)**.** *Pruning by Equation* (1) *(i.e., with $\varepsilon = 1$ in Line 7, Algorithm 1) returns exactly the same top-k set as fully evaluating all candidates, up to the underlying scorer's tie-breaking and floating-point semantics.*

The suffix correction $2\varepsilon\sqrt{R_q(m)R_x(m)}$ in Line 7, Algorithm 1, yields the guarantee with $\varepsilon = 1$; smaller values, $0 \le \varepsilon < 1$, prune earlier but may discard true neighbors, so we report them through throughput–recall frontiers. A true neighbor can be lost only if its unread residual is unusually aligned with the query's (Appendix B.4), hence moderate relaxations cost little recall.

Learning the basis. Equation (1) is valid by any orthogonal transform T; a particular T determines how much energy the suffix retains, and hence how early the bound becomes decisive. The choice of T thus constitutes an easy learning problem: minimize the expected suffix energy $\mathbb{E}[R_x(m)]$ over the orthogonal group, simultaneously for every prefix length m, from a sample of the data. By the Ky Fan variational principle (Horn & Johnson, 2012) this problem admits a closed-form solution: the PCA basis, ordered by decreasing eigenvalue, attains the minimum $\sum_{j>m}\lambda_j$ for all m at once (Appendix B.2). This optimality concerns the expected tail energy that the bound's slack depends on; it does not by itself imply the fewest coordinates read or the lowest runtime, which we measure (Section 5). The basis is fitted once, applied to stored vectors at build time and to queries at search time, and its quality is expressed by how steeply the fitted spectrum decays: if we model the normalized tail energy as $E(m) \simeq \exp(-\alpha m/d)$, a larger α tightens the bound earlier. Under the candidate-stream assumptions of Appendix B, a representative-margin model then predicts an expected coordinate work of

$$\mathbb{E}[\text{work}] \approx C_N N d/\alpha, \tag{2}$$

where N is the number of candidates and the slowly varying constant C_N is calibrated empirically. While this model prediction (Model prediction B.7) is distribution-dependent, and its constant is calibrated from measurement, it explains rather than predicts the ordering of our results: the fitted decay rate is $\alpha \in [8.3, 26.6]$ on the four datasets with $d \ge 768$ and only 3.6 and 6.9 on the two low-dimensional descriptors (Figure 7), hence the observed gains.

4 REALIZING THE BOUND INSIDE INDEXES

While Equation (1) is index-agnostic, the computational savings it entails are realized only if the pruning loop conforms with the way the index delivers its candidates. PANORAMA stores transformed

vectors in place of raw ones, adds a per-vector sidecar of the $L + 1$ suffix energies, and specializes the loop to three types of candidate stream as follows.

Contiguous lists. As inverted lists deliver long runs of candidates, applied on IVF-Flat, PANORAMA groups the vectors of each list into batches stored in level-major order: all candidates' coordinates for one level are stored contiguously before the next level, each level corresponds to one vectorized pass over the survivors, and the compute, bound-update, and survivor-compaction kernels are kept separate so that every hot loop vectorizes. The batch size trades threshold freshness against locality and the level count trades bound tightness against bookkeeping (Section 5). On IVFPQ, PANORAMA applies the same loop to codes, which it transposes within each level by their subquantizers so that one lookup-table slice stays hot while SIMD lanes fill across candidates. Nevertheless, raw PCA concentrates variance too aggressively for a uniform-bit PQ partition; therefore, for IVFPQ we *shape* the basis: we cap each level's energy at κ times the mean level energy and redistribute the surplus by orthogonal rotations followed by an intra-level rotation; the result is still orthogonal, so Equation (1) remains unchanged (Section 5 measures the effect on PQ recall).

Scattered candidates. Graph search and exact reranking expose noncontiguous IDs one at a time, so level-major batches are inapplicable; PANORAMA thus retains the native vector rows, holds the suffix energies in an ID-indexed sidecar, and dispatches underfilled candidate groups to the native exact kernel rather than paying bookkeeping. Inside the Faiss graph path the bound is tested against the current k^{th}-best result and a pruned node is still queued for expansion keyed by its partial distance, so the traversal may differ from exact HNSW and Proposition 3.1 applies to the candidates actually evaluated; in the fixed-graph kernel ablated in Section 5 we leave the traversal unchanged.

IVF–RaBitQ. RaBitQ scores sign codes using population counts (popcounts) after a random Haar rotation, which spreads energy evenly across coordinates, in contrast to PCA. Thus, a suffix-energy bound would be weak; the sign code, however, admits an exact combinatorial bound on the suffix contribution. PANORAMA keeps the `IndexIVFRaBitQ` (not the FastScan variant) with its lists, codes, parameters, and rotation unchanged and appends an eight-byte sidecar of cumulative sign popcounts to each code. The stored popcounts enable a *sign test* via a dot product; when $d > 768$, a gate first reads the leading 768 sign dimensions and bounds the unread suffix by the maximum contribution achievable by a sign pattern consistent with the stored popcounts. We omit candidates that cannot beat the running threshold and pass survivors to the unchanged RaBitQ scorer.

Proposition 4.1 (RaBitQ scorer identity)**.** *For a fixed IVF–RaBitQ index and query, the progressive variant returns bitwise-identical IDs and distances to the RaBitQ scorer.*

The exact-cardinality bound, its proof, the sidecar layout, and the identity checks are given in Appendix C.

5 EVALUATION

We investigate three questions:
Q1: does progressive verification make search faster for a given recall, across index families?
Q2: which design choices produce the gain, and what does each knob cost?
Q3: does the coordinate work that the cost model predicts turn into wall-clock speedup?

Protocol. Every measurement uses 1000 queries with $k = 10$, recall@10 against exact ground truth, one search thread, query transforms inside the timed region, and three complete wall-clock repetitions with no concurrent timed jobs, on one Intel Xeon Gold 5416S at 2.0 GHz with 128 GB DDR5-4400, Ubuntu 24.04, GCC 13.3, `-O3`, and AVX-512 (AVX2 for the IVF–RaBitQ pair). Frontiers sweep $nprobe$ or $efSearch$; every other knob is fixed per dataset (Appendix Table 5). The ε parameter is fixed per dataset; as we always report the measured recall, any recall loss due to a choice of $\varepsilon < 1$ is always visible in the reported results. Frontier figures show the mean queries-per-second (QPS) vs. mean recall@10 from 0.8 to 1.0; a curve that ends inside the window ends at its last measurement (full ranges in Appendix D); speedups are compared only at recalls both frontiers reach.

Datasets and baselines. The six datasets in Table 1 span 96 to 3072 dimensions. For full-precision IVF we compare against Faiss, PDX, ADSampling, DADE, and DDC; for compressed IVF, against

Table 1: Evaluation datasets.

Dataset	Size N	Size d	Semantics
Deep	9.99M	96	image emb.
SIFT	10.0M	128	image feat.
arXiv	2.24M	768	text emb.
GIST	1.00M	960	image feat.
Ada	0.90M	1536	text emb.
Large	0.90M	3072	text emb.

Table 2: Per-dataset IVF diagnostics.

Dataset	Dimensions evaluated: Pano.	DADE	DDC	ε	RaBitQ B/vec: Plain	Pano.
Deep	32.91%	36.65%	59.25%	1.0	92	100
SIFT	17.14%	25.85%	26.33%	1.0	116	124
arXiv	6.33%	6.71%	10.52%	0.7	692	700
GIST	4.25%	5.78%	4.03%	0.5	860	868
Ada	10.86%	5.71%	4.41%	0.8	1364	1372
Large	5.72%	2.88%	3.00%	0.7	2708	2716

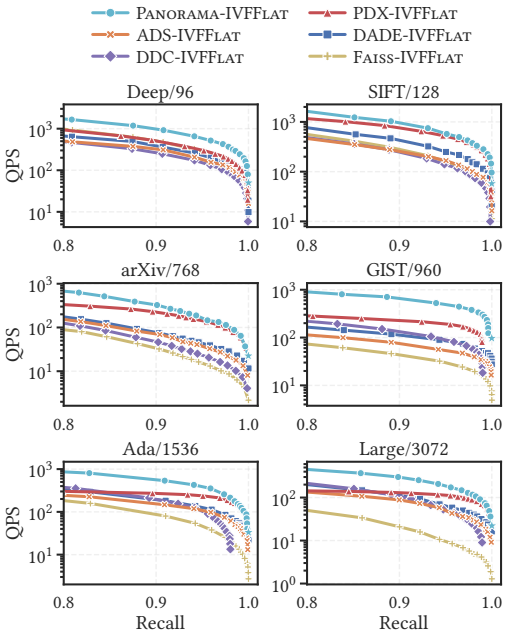


Figure 1: IVF-Flat QPS versus recall (window 0.8–1.0). The star marks PANORAMA's strict full-recall point. DDC is probabilistic and can plateau below the corresponding exact IVF frontier. PDX uses its ADSampling pruner.

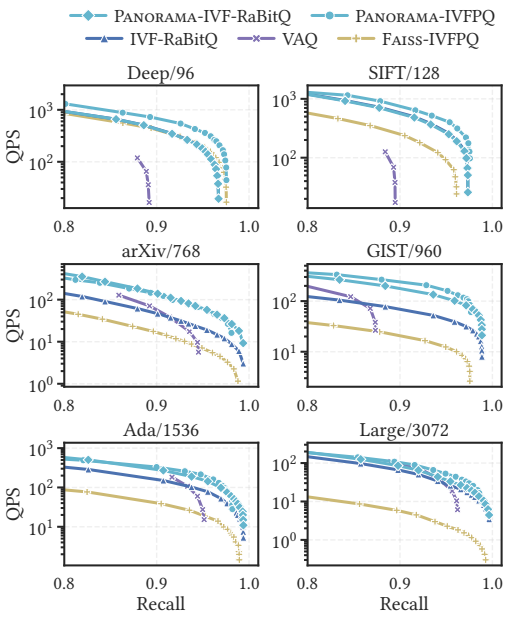


Figure 2: Compressed-index QPS versus recall (window 0.8–1.0). VAQ is compressed-only and memory-matched; RaBitQ is non-FastScan IVF–RaBitQ.

Faiss-IVFPQ, IVF–RaBitQ, and memory-matched VAQ; for graphs against Faiss-HNSW, ADSampling, and KS2 inside Faiss and against hnswlib and DADE as authors' systems; and for reranking against the Faiss FastScan reranker (see coverage in Appendix D, Table 4).

Clustered indexes (Q1). As Figure 1 shows, PANORAMA's IVF-Flat frontier lies above every baseline over the plotted window, on all six datasets, except near recall 0.99–1.0 on arXiv and Ada, where PDX is up to 10% faster; the separation grows with dimension and is largest on GIST, where the nearest baseline is at least $2.49\times$ slower across the window. Against PDX's deterministic BOND pruner, strict PANORAMA ($\varepsilon = 1$) is 1.8–$8.9\times$ faster at every matched recall from 0.80 to 0.99 on all six datasets. Table 2 gives the share of candidate coordinates evaluated at its smallest *nprobe* reaching recall@10 ≥ 0.95 (with Panorama's ε), for each method, together with the RaBitQ code bytes per vector. It shows that the fraction of coordinates read does not by itself determine throughput: DADE and DDC read fewer dimensions than PANORAMA on Ada and Large, yet PANORAMA is faster there, because its batching, layout, and kernel translate computational savings into time savings (see ablations below), while their statistical tests buy a weaker guarantee for that scan.

Compressed indexes (Q1). As Figure 2 shows, on Deep, arXiv, and Ada, where PANORAMA-IVFPQ and Faiss-IVFPQ use the same d-byte codes, PANORAMA-IVFPQ is at least $1.5\times$ faster than both Faiss-IVFPQ and VAQ across the window; on the other three datasets the code sizes differ and

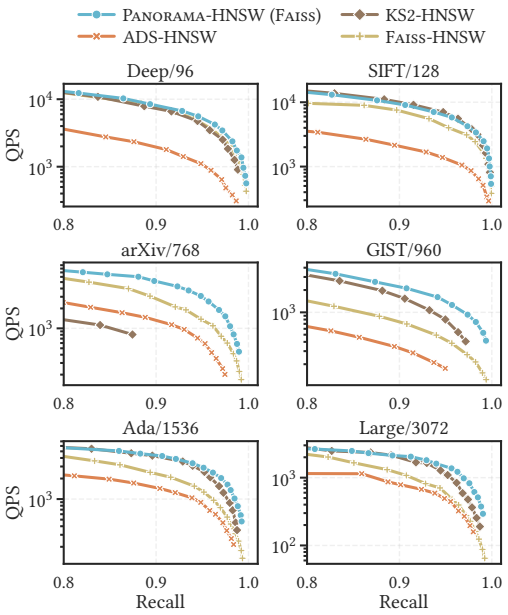


Figure 3: Faiss-family graph comparison (window 0.8–1.0). All but ADSampling share one graph per dataset; KS2's highest measured recall on arXiv is 0.874.

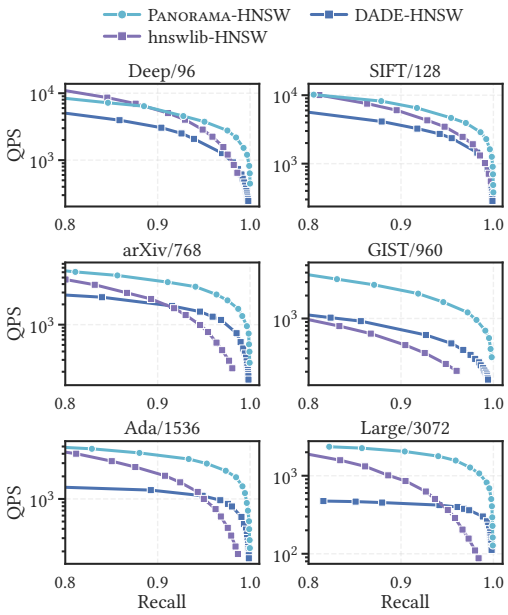


Figure 4: Authors-code graph comparison (window 0.8–1.0). Each system keeps its own graph construction; Appendix Figure 11 shows recall 0.6–1.0.

the comparison is not memory-matched. For IVF–RaBitQ, the PANORAMA variant changes QPS at recall 0.95 by $0.984\times$ on Deep, $0.962\times$ on SIFT, $3.006\times$ on arXiv, $2.588\times$ on GIST, $1.841\times$ on Ada, and $1.286\times$ on Large. Deep, SIFT, and arXiv have no prefix boundary, so their ratios isolate the stored popcount and dot-only sign kernel (a 2–4% cost at 96 and 128 dimensions, a gain at 768); on GIST, Ada, and Large the gate additionally rejects 82–99%, 4–50%, and under 1% of scanned candidates before their suffix bits are read (Appendix C), for eight extra bytes per vector and bitwise-identical results. VAQ improves with a larger code budget (see Appendix D, Figure 12).

Graph indexes (Q1). Figure 3 juxtaposes Faiss-family implementations on one shared graph per dataset (ADSampling excepted): PANORAMA-HNSW is the fastest method across the window on Deep, arXiv, and GIST, while KS2 is up to 9% faster on SIFT and slightly faster at the low end of the window on Ada and Large. Figure 4 compares end-to-end authors' systems whose graph construction differs (PANORAMA: $M = 32$, $efConstruction = 1000$; stock hnswlib: $M = 32$, $efConstruction = 40$; DADE: $M = 16$, $efConstruction = 500$, test calibrated on 100,000 distance pairs). At recall targets from 0.90 to 0.98, PANORAMA reaches 1.8–4.7$\times$ DADE's QPS on all six datasets ($3.36\times$ at GIST recall 0.98), while stock hnswlib is faster than PANORAMA on Deep at recall below 0.9, and does not reach recall 0.98 on GIST. These are systems comparisons: graph construction differs, so the gain over DADE combines verification with graph quality; the fixed-graph ablation (Figure 8c) isolates verification alone. We eschew a comparison vs. PEOs (Lu et al., 2024), which provides a *routing test* rather than a verification layer, adding metadata for every graph edge (4.98 GB against 4.15 GB on GIST) and requiring a long precomputation per edge, whereas PANORAMA sits on an unmodified index; the two strategies could well be combined.

Exact reranking (Q1). Figure 5 shows the results of applying PANORAMA to the exact reranking pass that follows a FastScan first stage, where a probabilistic test would defeat the purpose of the pass. From 768 dimensions upward PANORAMA is faster across the whole window, by at least 1.21–1.44$\times$ (the minimum occurs near recall 1), whereas on Deep the Faiss reranker is up to 6% faster between recall 0.94 and 0.99.

Runtime knobs (Q2). Figure 6 and Appendix D, Figure 10 sweep the four knobs on Ada with every list probed. A tighter shaping cap κ raises PQ recall slightly but costs throughput, since a flatter

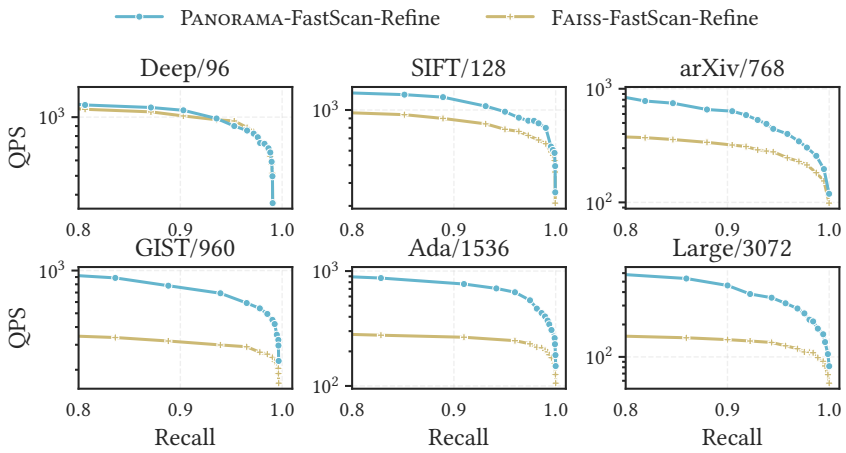


Figure 5: FastScan with exact full-precision reranking (recall window 0.8–1.0).

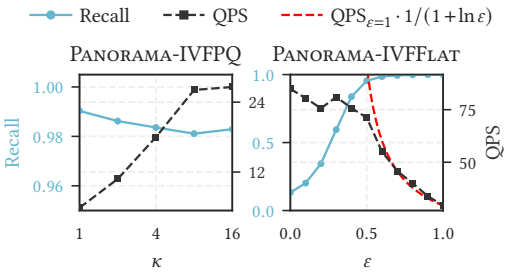


Figure 6: Pruning knobs on Ada with every list probed. Left: PANORAMA-IVFPQ recall and QPS versus the shaping cap κ. Right: PANORAMA-IVFFlat versus ε, with the predicted $\mathrm{QPS}_{\varepsilon=1}/(1+\ln\varepsilon)$ (dashed).

spectrum loosens the bound. Relaxing ε keeps recall near 1 down to $\varepsilon = 0.5$ and loses it quickly below, while throughput follows the $1/(1+\ln\varepsilon)$ scaling of Appendix B.4 for $\varepsilon \geq 0.6$. Additionally, increasing levels tightens the bound but adds bookkeeping, and enlarging batches improves locality at the cost of a staler threshold; both have a broad plateau around the selected setting.

Why a learned basis (Q2). Figure 7 compares the tail energy left by three orthogonal transforms and the raw basis on each dataset: PCA leaves the least at every prefix, the fitted exponential tracks it with mean absolute error below 0.04, and shaping flattens the spectrum only as much as the cap demands. Figure 8a shows how spectra translate to time on GIST: a raw-space IVF-Flat index supplies one fixed candidate stream to every arm, and a common strict scanner times the query transform plus the scan. PCA reaches $1.92\times$ the raw-order throughput, against $1.21\times$ for DCT-II and $1.19\times$ for balanced Haar, with every arm returning the exhaustive top-10 of its candidates (recall@10 $= 0.8767$).

Why the layout and the kernel (Q2). Figure 8b presents the results obtained by adding the system components of Section 4 one at a time with the index and candidate stream fixed (IVF128 with all lists probed, 30 levels, strict $\varepsilon = 1$): over a point-wise progressive scan, batching with a level-major layout gives $9.27\times$, the optimized kernel a further $1.46\times$, and fixed-width specialization $1.034\times$, for $14.00\times$ overall. All four stages return the same top-10 sets and recall@10 $= 1.0$ (A3 swaps the order of two results for one query), hence the gain is in how the bound is evaluated, not in what it decides. Figure 8c performs such an ablation on scattered candidates: it holds the GIST graph ($M = 32$, $efConstruction = 1000$), the transformed rows, and the eight-level sidecar fixed and changes only the level-0 distance kernel. Strict PANORAMA returns the same ordered IDs and recall@10 as exact evaluation with $1.43\times$ the mean QPS at $efSearch = 64$ and $1.53\times$ at $efSearch = 256$, so even a point-wise scattered kernel retains computational savings.

Cost model (Q3). The cost model of Section 3 models the fraction p of evaluated coordinates; Appendix B.3, Figure 9 tests the assumption that the saved coordinate work becomes wall-clock speedup, by plotting measured speedup against $1/p$ from the same run at up to five recall targets per dataset, with no fitted parameter. IVFFlat points lie on or below the diagonal: bookkeeping absorbs

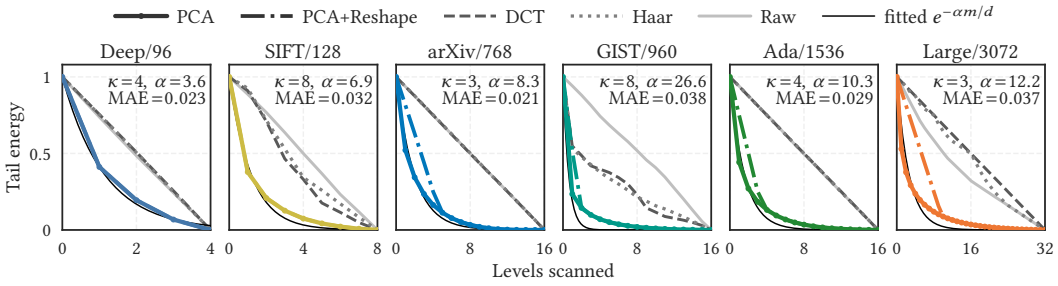
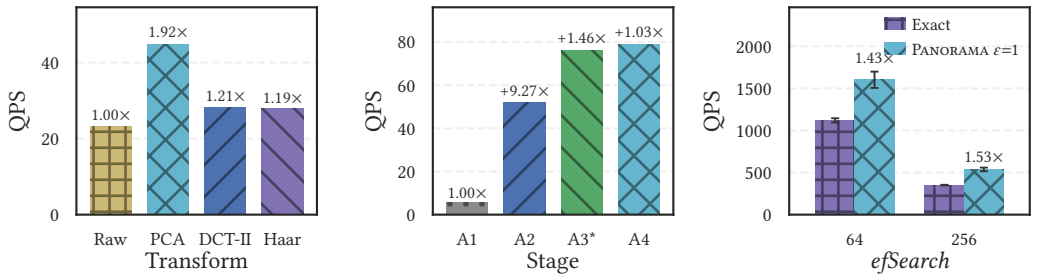


Figure 7: Tail energy per level for PCA (solid) and PQ-compatible PCA+Reshape (dash-dot) against DCT (dashed), Haar (dotted), and Raw (light gray), with the exponential fitted to PCA as a thin black line; on Deep and SIFT, PCA+Reshape coincides with PCA.



(a) Transform (b) Systems stages (c) Fixed-graph HNSW

Figure 8: GIST ablations; 1,000 queries, $k = 10$, one thread, query transform included. (a) Median transform-plus-scan QPS on one fixed IVF128/$nprobe$=8 candidate stream. (b) A1 point-wise scan, A2 batched level-major scan, A3 optimized kernel, A4 fixed-width specialization; labels give the gain over the previous stage and * marks A3's within-set rank swap. (c) Exact versus strict PANORAMA level-0 distance evaluation on the same graph; error bars show one standard deviation.

part of the saved work and the rest reaches the clock. IVFPQ points spread on both sides by up to $4\times$, so its speedup is not explained by coordinate work alone.

Limitations. We recognize that progressive bookkeeping can outweigh the saved computations in low dimensions, where spectra decay gradually: RaBitQ, shared-graph HNSW, and reranking show small or negative gains at 96 and 128 dimensions. Relaxed ε settings, just like DDC's probabilistic test, may lower the maximum reachable recall, while PCA fitting, calibration, graph construction, and ISA-specific kernels add costs that QPS alone does not show. All timings are single-threaded on one AVX-512 machine, the three ablations use GIST only, and the cost model's constant is calibrated rather than derived. The PQ comparison is memory-matched only on Deep, arXiv, and Ada, and the RaBitQ integration targets the non-FastScan scorer.

6 CONCLUSION

Candidate verification in nearest-neighbor search grows with embedding dimension. PANORAMA renders that step progressive: we use a PCA basis learned from the data to orders the coordinates so that a Cauchy–Schwarz bound decides most candidates after reading a small fraction of their coordinates, exactly in the strict mode and with one knob for trading recall against speed. The saving is a property of learned representations, since the more steeply an embedding's spectrum decays in the fitted basis, the earlier its candidates can be rejected, and it is largest on the wide text embeddings that retrieval-augmented systems search. Three directions follow form our work. Embeddings could be trained for cheap verification, with objectives that reward steep spectral decay or nested structure, so that retrieval cost becomes part of the representation-learning objective. The bound extends to inner-product and cosine search through the usual reductions to Euclidean distance, and to updatable collections through incremental estimates of the basis. Lastly. the kernels can be carried to multi-threaded, GPU, and disk-resident settings, and combined with the graph-routing tests that decide which neighbors to visit.

Ethics Statement

This work accelerates nearest-neighbor search over existing public datasets of image descriptors and text embeddings; it collects no new data and involves no human subjects. Faster retrieval lowers the cost of search and retrieval-augmented systems, which also lowers the cost of deploying them for harmful ends; the method does not change what such systems retrieve, and in its strict mode it returns exactly the results of the underlying index.

Reproducibility Statement

Section 5 states the measurement protocol, hardware, and how every non-swept parameter was chosen, and Appendix Table 5 lists the per-dataset index parameters. Appendix B gives the proofs and states every modeling assumption of the cost model, Appendix C gives the IVF–RaBitQ layout, bound, identity checks, and search grids, and Appendix D lists the measured recall range of every plotted frontier. The implementation, the baseline integrations, the benchmark configurations, and the plotting scripts are available at `https://anonymous.4open.science/r/panorama-iclr27`.

LLM Usage Statement

Large language models were used to assist with editing the text, formatting tables and figures, and writing plotting scripts. All methods, experiments, results, and claims were designed, run, and checked by the authors, who take full responsibility for the content.

References

Fabien André, Anne-Marie Kermarrec, and Nicolas Le Scouarnec. Cache locality is not enough: High-performance nearest neighbor search with product quantization fast scan. *Proc. VLDB Endow.*, 9(4):288–299, 2015.

Artem Babenko and Victor S. Lempitsky. The inverted multi-index. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(6):1247–1260, 2015.

Patrick H. Chen, Wei-Cheng Chang, Jyun-Yu Jiang, Hsiang-Fu Yu, Inderjit S. Dhillon, and Cho-Jui Hsieh. FINGER: fast inference for graph-based approximate nearest neighbor search. In *ACM Web Conference*, pp. 3225–3235, 2023.

Yunchao Chen, Tao Guan, and Cheng Wang. Approximate nearest neighbor search by residual vector quantization. In *Sensors*, volume 10, pp. 11259–11273. MDPI, 2010.

Arjen P. de Vries, Nikos Mamoulis, Niels Nes, and Martin Kersten. Efficient k-nn search on vertically decomposed data. In *Proceedings of the 2002 ACM SIGMOD International Conference on Management of Data*, SIGMOD '02, pp. 322–333, New York, NY, USA, 2002. Association for Computing Machinery. ISBN 1581134975. doi: 10.1145/564691.564729. URL `https://doi.org/10.1145/564691.564729`.

Liwei Deng, Penghao Chen, Ximu Zeng, Tianfu Wang, Yan Zhao, and Kai Zheng. Efficient data-aware distance comparison operations for high-dimensional approximate nearest neighbor search, 2024. URL `https://arxiv.org/abs/2411.17229`.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. The faiss library. 2025.

Kawin Ethayarajh. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP-IJCNLP)*, pp. 55–65, 2019.

Cong Fu, Chao Xiang, Changxu Wang, and Deng Cai. Fast approximate nearest neighbor search with the navigating spreading-out graph. In *Proceedings of the VLDB Endowment*, volume 12, pp. 461–474. VLDB Endowment, 2019.

Jianyang Gao and Cheng Long. High-dimensional approximate nearest neighbor search: with reliable and efficient distance comparison operations. *Proc. ACM Manag. Data*, 1(2):137:1–137:27, 2023.

Jianyang Gao and Cheng Long. RaBitQ: Quantizing high-dimensional vectors with a theoretical error bound for approximate nearest neighbor search. *Proc. ACM Manag. Data*, 2(3):167, 2024.

Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tie-Yan Liu. Representation degeneration problem in training natural language generation models. In *International Conference on Learning Representations (ICLR)*, 2019.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2023.

Tiezheng Ge, Kaiming He, Qifa Ke, and Jian Sun. Optimized product quantization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(4):744–755, 2013.

Ruiqi Guo, Philip Sun, Erik Lindgren, Quan Geng, David Simcha, Felix Chern, and Sanjiv Kumar. Accelerating large-scale inference with anisotropic vector quantization. *Proceedings of the 37th International Conference on Machine Learning (ICML)*, pp. 3887–3896, 2020.

Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2nd edition, 2012.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547, 2019.

William B. Johnson and Joram Lindenstrauss. Extensions of lipschitz mappings into a hilbert space. *Contemporary Mathematics*, 26(189-206):1–1, 1984.

H. Jégou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):117–128, 2010.

Hervé Jégou, Matthijs Douze, and Cordelia Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *European Conference on Computer Vision (ECCV)*, pp. 304–317. Springer, 2008.

Hervé Jégou, Romain Tavenard, Matthijs Douze, and Laurent Amsaleg. Searching in one billion vectors: re-rank with source coding. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 861–864. IEEE, 2011.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781, 2020.

Shrikant Kashyap and Panagiotis Karras. Scalable kNN search on vertically stored time series. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1334–1342, 2011. ISBN 9781450308137. URL https://doi.org/10.1145/2020408.2020607.

Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

Leonardo X. Kuffo, Elena Krippner, and Peter A. Boncz. PDX: A data layout for vector similarity search. *Proc. ACM Manag. Data*, 3(3):196:1–196:26, 2025. doi: 10.1145/3725333. URL https://doi.org/10.1145/3725333.

Aditya Kusupati, Gantavya Bhatt, Aniket Rege, Matthew Wallingford, Aditya Sinha, Vivek Ramanujan, William Howard-Snyder, Kaifeng Chen, Sham Kakade, Prateek Jain, et al. Matryoshka representation learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 30233–30249, 2022.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474, 2020.

Kejing Lu, Chuan Xiao, and Yoshiharu Ishikawa. Probabilistic routing for graph-based approximate nearest neighbor search. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Kejing Lu, Chuan Xiao, and Yoshiharu Ishikawa. Probabilistic kernel function for fast angle testing. *CoRR*, abs/2505.20274, 2025.

Yu A. Malkov and Dmitry A. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(4):824–836, 2020.

Julieta Martinez, Joris Clement, Holger H. Hoos, and James J. Little. Revisiting additive quantization. In *European Conference on Computer Vision (ECCV)*, pp. 137–153. Springer, 2016.

Pascal Massart. The tight constant in the dvoretzky–kiefer–wolfowitz inequality. *The Annals of Probability*, 18(3):1269–1283, July 1990. doi: 10.1214/aop/1176990746.

Jiaqi Mu and Pramod Viswanath. All-but-the-top: Simple and effective postprocessing for word representations. In *International Conference on Learning Representations (ICLR)*, 2018.

Arvind Neelakantan, Tao Xu, Raul Puri, Alec Radford, Jesse Michael Han, Jerry Tworek, Qiming Yuan, Nikolas Tezak, Jong Wook Kim, Chris Hallacy, Johannes Heidecke, Pranav Shyam, Boris Power, Tyna Eloundou Nekoul, Girish Sastry, Gretchen Krueger, David Schnurr, Felipe Petroski Such, Kenny Hsu, Madeleine Thompson, Tabarak Khan, Toki Sherbakov, Joanne Jang, Peter Welinder, and Lilian Weng. Text and code embeddings by contrastive pre-training, 2022. URL `https://arxiv.org/abs/2201.10005`.

OpenAI. Embeddings. `https://platform.openai.com/docs/guides/embeddings`. Accessed: 2025-09-22.

James Jie Pan, Jianguo Wang, and Guoliang Li. Survey of vector database management systems. *The VLDB Journal*, 33(5):1591–1615, 2024.

John Paparrizos, Ikraduya Edian, Chunwei Liu, Aaron J. Elmore, and Michael J. Franklin. Fast adaptive similarity search through variance-aware quantization. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pp. 2969–2983, 2022. doi: 10.1109/ICDE53745.2022.00268.

Suhas Jayaram Subramanya, Fnu Devvrit, Harsha Vardhan Simhadri, Ravishankar Krishnaswamy, and Rohan Kadekodi. Diskann: Fast accurate billion-point nearest neighbor search on a single node. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 511–518, 2001.

Lidan Wang, Jimmy Lin, and Donald Metzler. A cascade ranking model for efficient ranked retrieval. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 105–114, 2011.

Zeyu Wang, Haoran Xiong, Qitong Wang, Zhenying He, Peng Wang, Themis Palpanas, and Wei Wang. Dimensionality-reduction techniques for approximate nearest neighbor search: A survey and evaluation, 2025. URL `https://arxiv.org/abs/2403.13491`.

Roger Weber, Hans-Jörg Schek, and Stephen Blott. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. In *VLDB*, volume 98, pp. 194–205, 1998.

Haike Xu, Sandeep Silwal, and Piotr Indyk. A bi-metric framework for fast similarity search. *CoRR*, abs/2406.02891, 2024.

Mingyu Yang, Wentao Li, Jiabao Jin, Xiaoyao Zhong, Xiangyu Wang, Zhitao Shen, Wei Jia, and Wei Wang. Effective and general distance computation for approximate nearest neighbor search. In *2025 IEEE 41st International Conference on Data Engineering (ICDE)*, pp. 1098–1110. IEEE, 2025.

A Appendix Overview

Appendix B proves strict-mode safety and PCA optimality and states the assumptions and derivations of the cost model of Section 3 and of the ε-relaxation of Section 5. Appendix C details the IVF–RaBitQ integration, and Appendix D lists the measured recall range of every frontier.

Contents of the Appendix

B Bound Safety and Cost Model **13**

B.1 Notation and strict-mode safety 13

B.2 PCA minimises expected tail energy 14

B.3 Calibrated representative-margin cost model 15

B.4 The ε-relaxation 18

C IVF–RaBitQ Integration **20**

C.1 Sign test and exact-cardinality gate 20

C.2 Boundaries, sidecar layout, and survivor scoring 21

C.3 Proof of Proposition 4.1 and identity checks 21

C.4 Experimental parameters 22

D Recall Coverage and Parameters **22**

B Bound Safety and Cost Model

This appendix separates exact results from a cost model. The exact results are the validity of the Cauchy–Schwarz lower bound with strict-mode safety (Lemma B.1, which proves Proposition 3.1), the optimality of PCA for expected tail energy (Section B.2, stated informally in Section 3), and the gap identity for the ε-relaxed bound (Lemma B.8). Lemma B.3 (DKW) is also exact, but only under the i.i.d. sampling model of Assumption A3. The cost model is a *calibrated representative-margin model*: under the modelling assumptions A1–A5 it predicts the fraction of coordinates PANORAMA evaluates (Model prediction B.7) and the effect of the ε-relaxation (Heuristic prediction B.10). Statements labelled Model, Model prediction, or Heuristic prediction use $\approx$ for equality under these assumptions; they are approximations, not bounds, and their constant is calibrated from measurement. Lemmas, definitions, models, and predictions share one counter prefixed by this appendix's letter; assumptions are labelled A1–A5.

B.1 Notation and strict-mode safety

We adopt the notation of the main paper. All vectors lie in $\mathbb{R}^d$. For an orthogonal transform T and a level $0 \le m \le d$, the *tail energy* of a vector $\mathbf{v}$ is

$$R^{(m,d)}_{T(\mathbf{v})} := \sum_{j=m+1}^{d} T(\mathbf{v})_j^2.$$

The level-m Cauchy–Schwarz lower bound on $\|\mathbf{q}-\mathbf{x}\|^2$ is

$$\mathsf{LB}^m(\mathbf{q},\mathbf{x}) = R^{(0,d)}_{T(\mathbf{q})} + R^{(0,d)}_{T(\mathbf{x})} - 2\left(p^{(0,m)}(\mathbf{q},\mathbf{x}) + \sqrt{R^{(m,d)}_{T(\mathbf{q})}\, R^{(m,d)}_{T(\mathbf{x})}}\right), \tag{3}$$

where $p^{(0,m)}(\mathbf{q},\mathbf{x}) = \sum_{j=1}^{m} T(\mathbf{q})_j T(\mathbf{x})_j$. The ε-relaxed bound replaces the second factor by $2\varepsilon\sqrt{R^{(m,d)}_{T(\mathbf{q})} R^{(m,d)}_{T(\mathbf{x})}}$, with $\varepsilon = 1$ recovering (3).

PANORAMA maintains a running threshold τ, the squared distance of the k^{th} nearest neighbour found so far (written d_k in the main paper). A candidate $\mathbf{x}$ is pruned at level m if $\mathsf{LB}^m(\mathbf{q}, \mathbf{x}) > \tau$, equivalently if the gap

$$\|\mathbf{q} - \mathbf{x}\|^2 - \mathsf{LB}^m(\mathbf{q}, \mathbf{x}) < \Delta \;:=\; \|\mathbf{q} - \mathbf{x}\|^2 - \tau.$$

We call Δ the *pruning margin*. In strict mode ($\varepsilon = 1$), true k-nearest neighbours satisfy $\Delta \leq 0$ and are never pruned by a valid lower bound. Lemma B.1 makes this precise.

Lemma B.1 (Bound validity and strict-mode safety)**.** *For every orthogonal T, level m, and pair $(\mathbf{q}, \mathbf{x})$, $\mathsf{LB}^m(\mathbf{q}, \mathbf{x}) \leq \|\mathbf{q} - \mathbf{x}\|^2$. Consequently, in exact arithmetic and with $\varepsilon = 1$, every pruned candidate is strictly farther from $\mathbf{q}$ than the k^{th} nearest supplied candidate, and* PANORAMA *returns the same top-k set as exact scoring of every supplied candidate, up to the scorer's tie-breaking.*

Proof. Orthogonality gives $\|\mathbf{q} - \mathbf{x}\|^2 = \|T(\mathbf{q}) - T(\mathbf{x})\|^2 = R^{(0,d)}_{T(\mathbf{q})} + R^{(0,d)}_{T(\mathbf{x})} - 2p^{(0,m)}(\mathbf{q}, \mathbf{x}) - 2\pi_m$ with $\pi_m = \sum_{j>m} T(\mathbf{q})_j T(\mathbf{x})_j$, and Cauchy–Schwarz on the residual coordinates gives $\pi_m \leq \sqrt{R^{(m,d)}_{T(\mathbf{q})} R^{(m,d)}_{T(\mathbf{x})}}$, hence the bound. Let $d^\star_k$ be the k^{th} smallest exact squared distance over all supplied candidates. At every step, including with batched or stale thresholds, τ is the k^{th} smallest exact distance over a subset of fully evaluated candidates (or $+\infty$), so $\tau \geq d^\star_k$. A pruned candidate therefore satisfies $\|\mathbf{q} - \mathbf{x}\|^2 \geq \mathsf{LB}^m(\mathbf{q}, \mathbf{x}) > \tau \geq d^\star_k$, so every candidate at distance at most $d^\star_k$ is fully evaluated and the top-k of the evaluated set equals that of the supplied set. □

B.2 PCA MINIMISES EXPECTED TAIL ENERGY

Lemma B.2 (Ky Fan)**.** *Let $\Sigma \in \mathbb{R}^{d\times d}$ be symmetric positive semidefinite with eigenvalues $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$. For every $m \in \{1, \ldots, d\}$,*

$$\max_{T \in \mathrm{O}(d)} \sum_{j=1}^{m} (T\Sigma T^\top)_{jj} \;=\; \sum_{j=1}^{m} \lambda_j,$$

with the maximum attained when the first m rows of T span the eigenspace associated with the top-m eigenvalues of Σ.

Proof (sketch). The diagonal of $T\Sigma T^\top$ is majorised by the eigenvalues of Σ (Schur–Horn theorem). For any m, the partial sum of the m largest diagonal entries is at most the partial sum of the m largest eigenvalues of Σ. Equality is attained by the orthogonal transform whose first m rows span the corresponding eigenspace. See Horn & Johnson (2012, Section 4.3) for the full proof. □

Proposition (PCA energy-optimality; cf. Section 3)**.** *Let $X \in \mathbb{R}^d$ be a centred random vector ($\mathbb{E}[X] = \mathbf{0}$) with covariance $\Sigma = \mathbb{E}[XX^\top]$ having eigenvalues $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$. For any orthogonal $T \in \mathrm{O}(d)$ and any $m \in \{1, \ldots, d-1\}$,*

$$\mathbb{E}\Big[R^{(m,d)}_{T(X)}\Big] \;\geq\; \sum_{j=m+1}^{d} \lambda_j,$$

with equality when the rows of T are eigenvectors of Σ ordered by decreasing eigenvalue.

Proof. For any orthogonal T,

$$\mathbb{E}\big[(TX)_j^2\big] \;=\; \mathbb{E}\big[(TX)(TX)^\top\big]_{jj} \;=\; \big(T\,\mathbb{E}[XX^\top]\,T^\top\big)_{jj} \;=\; (T\Sigma T^\top)_{jj}.$$

Summing over $j > m$,

$$\mathbb{E}\Big[R^{(m,d)}_{T(X)}\Big] \;=\; \sum_{j=m+1}^{d} (T\Sigma T^\top)_{jj}.$$

Orthogonal conjugation preserves the trace, so

$$\sum_{j=1}^{d} (T\Sigma T^\top)_{jj} \;=\; \mathrm{Tr}(T\Sigma T^\top) \;=\; \mathrm{Tr}(\Sigma) \;=\; \sum_{j=1}^{d} \lambda_j.$$

Therefore

$$\mathbb{E}\Big[R^{(m,d)}_{T(X)}\Big] \;=\; \sum_{j=1}^{d}\lambda_j \;-\; \sum_{j=1}^{m}(T\Sigma T^\top)_{jj}.$$

By Lemma B.2,

$$\sum_{j=1}^{m}(T\Sigma T^\top)_{jj} \;\le\; \sum_{j=1}^{m}\lambda_j,$$

with equality when the first m rows of T span the top-m eigenspace of Σ. Substituting,

$$\mathbb{E}\Big[R^{(m,d)}_{T(X)}\Big] \;\ge\; \sum_{j=1}^{d}\lambda_j \;-\; \sum_{j=1}^{m}\lambda_j \;=\; \sum_{j=m+1}^{d}\lambda_j,$$

with equality whenever the first m rows of T span a top-m eigenspace, in particular for the PCA basis. □

B.3 Calibrated representative-margin cost model

This section models the refinement work of PANORAMA under the Gaussian candidate-stream model. It has three ingredients: a sampling model for the running k^{th} threshold, a representative margin that replaces each candidate's own distance by the mean distance μ, and the Cauchy–Schwarz pruning depth at that margin. The second ingredient is a modelling step, not a derivation: it yields no candidate-wise margin, so the resulting cost expressions are predictions whose constant is calibrated from measurement, not upper bounds.

Modelling assumptions. The cost model rests on the following assumptions; Lemmas B.1 and B.8 and the PCA result use none of them.

Assumption A1 (Exponential PCA tail energy)**.** We take T to be the PCA basis of the dataset. On real embeddings its tail energy decays exponentially,

$$R^{(m,d)}_{T(\mathbf{x})} \;\approx\; \|\mathbf{x}\|^2\, e^{-\alpha m/d},$$

where $\alpha > 0$ is the per-dataset decay rate, fitted by least squares to the dataset's normalised PCA tail-energy curve. This exponential profile is an empirical property of the PCA basis, which we verify directly: the mean absolute fit error of the model is below 0.04 on every dataset (Figure 7). The $\approx$ records exactly this finite-sample modelling claim; applying the dataset-level fit to each individual vector is part of the model.

Assumption A2 (Bounded norms)**.** There is a constant R such that $\|T(\mathbf{x})\| \le R$ for every $\mathbf{x}$ in the dataset and every query $\mathbf{q}$. We write $C_0 := 4R^2$.

Assumption A3 (Gaussian distance model)**.** For a fixed query $\mathbf{q}$, the squared distances $\|\mathbf{q}-\mathbf{x}\|^2$ in the candidate stream are modelled as i.i.d. draws from a Gaussian with mean μ and standard deviation σ. The exact distribution is chi-square-like and the candidate stream produced by an ANN index is not literally i.i.d.; the Gaussian order-statistic calculation is the tractable cost model whose constant is calibrated empirically.

Assumption A4 (One dimension per level)**.** The model reasons at the finest pruning granularity, $L = d$, $m_\ell = \ell$.

A fifth assumption (residual isotropy, A5) is introduced and used only in Section B.4.

PANORAMA maintains a pruning threshold τ, the squared distance of the k^{th} nearest neighbour found so far. For analytical tractability we model τ_i as the k^{th} order statistic among i i.i.d. draws from the distance distribution, acknowledging that the algorithm's threshold actually arises from a mixture of exact and pruned candidates.

Lemma B.3 (High-probability bound on the sampled threshold via DKW)**.** *Let the squared distances be i.i.d. with CDF F. For any $\epsilon \in (0,1)$, with probability at least $1 - 2e^{-2i\epsilon^2}$ (Dvoretzky–Kiefer–Wolfowitz; Massart, 1990), the k^{th} order statistic τ_i satisfies*

$$F^{-1}\Big(\max\big\{0, \tfrac{k}{i+1} - \epsilon\big\}\Big) \;\le\; \tau_i \;\le\; F^{-1}\Big(\min\big\{1, \tfrac{k}{i+1} + \epsilon\big\}\Big).$$

Under the Gaussian assumption A3 ($F(r) = \Phi((r-\mu)/\sigma)$), this gives in particular

$$\tau_i \;\le\; \mu + \sigma\,\Phi^{-1}\Big(\tfrac{k}{i+1} + \epsilon\Big) \qquad \textit{with probability at least } 1 - 2e^{-2i\epsilon^2}.$$

Proof. Let F_i be the empirical CDF of the first i distances. The DKW inequality gives $\Pr(\sup_t |F_i(t) - F(t)| > \epsilon) \le 2e^{-2i\epsilon^2}$. On the event $\sup_t |F_i - F| \le \epsilon$ we have $F(t) - \epsilon \le F_i(t) \le F(t) + \epsilon$ for all t; monotonicity of F^{-1} then gives $F^{-1}(u-\epsilon) \le F_i^{-1}(u) \le F^{-1}(u+\epsilon)$ for all $u \in (0,1)$. Taking $u = k/(i+1)$ and recalling $\tau_i = F_i^{-1}(k/(i+1))$ yields the two-sided bound. Under A3, $F^{-1}(p) = \mu + \sigma\,\Phi^{-1}(p)$, which gives the Gaussian form. □

Lemma B.3 controls only the sampled threshold τ_i, and only under the i.i.d. distance model of Assumption A3. It says nothing about the distance of any particular candidate and therefore supplies no candidate-level pruning guarantee.

A new candidate is tested against τ_i. Its own squared distance is random under A3; the model replaces it by the mean μ, which defines a representative margin.

Definition B.4 (Representative margin Δ_i)**.** Fix $\epsilon_i \in (0,1)$ and set the sampled k^{th}-NN threshold upper bound

$$K_i := \mu + \sigma\,\Phi^{-1}\Big(\tfrac{k}{i+1} + \epsilon_i\Big), \qquad \Delta_i := \mu - K_i = -\sigma\,\Phi^{-1}\Big(\tfrac{k}{i+1} + \epsilon_i\Big).$$

On the DKW event of Lemma B.3, $\tau_i \le K_i$, so a candidate at squared distance exactly μ would have margin at least Δ_i. We use Δ_i as the per-step representative margin in the cost model; candidates closer than μ have smaller margins and farther ones larger margins. The model does not track this spread; its net effect on cost is absorbed into the constant C_N below, which is calibrated empirically. The margin is positive whenever $\frac{k}{i+1} + \epsilon_i < \frac{1}{2}$ (so $\Phi^{-1}(\cdot) < 0$).

Uniform threshold schedule. Fix a target failure probability $\delta \in (0,1)$ and set

$$\epsilon_i := \sqrt{\tfrac{1}{2i}\,\ln\tfrac{2N}{\delta}}.$$

By a union bound over $i \in \{k+1, \dots, N\}$, the event

$$\mathcal{E}_\delta := \bigcap_{i=k+1}^{N} \Big\{\tau_i \le \mu + \sigma\,\Phi^{-1}\big(\tfrac{k}{i+1} + \epsilon_i\big)\Big\}$$

holds with probability at least $1-\delta$. The model below is evaluated on $\mathcal{E}_\delta$, an event about the threshold sequence only.

A candidate $\mathbf{x}_j$ processed at step i is pruned at level m exactly when its gap is below its own margin $\|\mathbf{q} - \mathbf{x}_j\|^2 - \tau_i$. The model replaces that margin by the representative margin Δ_i:

$$\|\mathbf{q} - \mathbf{x}_j\|^2 - \mathsf{LB}^m(\mathbf{q}, \mathbf{x}_j) \;<\; \Delta_i.$$

By Cauchy–Schwarz the left-hand side is at most four times the geometric mean of the tail energies; applying Assumptions A1 and A2,

$$4\sqrt{R^{(m,d)}_{T(\mathbf{q})}\,R^{(m,d)}_{T(\mathbf{x}_j)}} \approx 4\sqrt{\|\mathbf{q}\|^2\|\mathbf{x}_j\|^2\,e^{-2\alpha m/d}} \;\le\; C_0\,e^{-\alpha m/d}, \qquad C_0 := 4R^2.$$

In the model, pruning therefore occurs once $C_0\,e^{-\alpha m/d} < \Delta_i$, i.e.

$$e^{-\alpha m/d} < \frac{\Delta_i}{C_0} \iff m > \frac{d}{\alpha}\,\ln\frac{C_0}{\Delta_i}.$$

Model B.5 (Representative pruning depth)**.** The modelled number of dimensions ρ_i processed for the candidate at step i is

$$\rho_i \;\approx\; \min\left\{d,\; \frac{d}{\alpha}\left[\ln\frac{C_0}{\Delta_i}\right]_+\right\}, \qquad C_0 = 4R^2, \quad [x]_+ := \max\{0, x\}.$$

The total cost is dominated by the sum of the pruning dimensions over the N candidates. Writing $i_0 := \min\{i \geq k+1 : \frac{k}{i+1} + \epsilon_i < \frac{1}{2}\}$ for the first index at which the model margin becomes positive, and neglecting the at most $i_0 - 1$ earlier candidates of cost at most d each,

$$\text{Cost} = \sum_{i=k+1}^{N} \rho_i \;\approx\; \sum_{i=\max\{i_0,k+1\}}^{N} \frac{d}{\alpha} \min\Big\{\alpha, \Big[\ln \frac{C_0}{\Delta_i}\Big]_+\Big\}.$$

Let $I_{C_0} := \{i \in \{\max\{i_0, k+1\}, \ldots, N\} : \Delta_i \leq C_0\}$ and $N_{C_0} := \max I_{C_0}$. Then

$$\sum_i \Big[\ln \frac{C_0}{\Delta_i}\Big]_+ = \sum_{i \in I_{C_0}} (\ln C_0 - \ln \Delta_i) = |I_{C_0}| \, \ln C_0 - \ln\Big(\prod_{i \in I_{C_0}} \Delta_i\Big).$$

Equivalently, define the finite-sample model constant

$$C_N \;:=\; \frac{1}{N} \sum_{i=\max\{i_0,k+1\}}^{N} \min\Big\{\alpha, \Big[\ln \frac{C_0}{\Delta_i}\Big]_+\Big\}. \tag{4}$$

This is the average capped log-margin cost per candidate.

Model B.6 (Cost via averaged margins)**.** The total computational cost is modelled by

$$\text{Cost} \;\approx\; \frac{C_N}{\alpha} Nd.$$

When the cap at d is inactive on the contributing window, this is equivalently

$$C_N \;\approx\; \frac{1}{N}\Big(|I_{C_0}| \, \ln C_0 - \ln \prod_{i \in I_{C_0}} \Delta_i\Big).$$

The model constant is the average in Model B.6. To see its dependence on the Gaussian model parameters, write $p_i := k/(i+1) + \epsilon_i$. When p_i lies in the lower tail, the usual Gaussian-quantile approximation gives

$$\Delta_i = -\sigma \, \Phi^{-1}(p_i) \;\approx\; \sigma\sqrt{2\ln(1/p_i)}. \tag{5}$$

Substituting (5) into the average shows that the constant has the representative form

$$C_N \;\approx\; \frac{|I_{C_0}|}{N} \ln \frac{C_0}{\sigma} \;-\; \zeta_N \quad \text{(with the cap at } \alpha \text{ applied candidate-wise)},$$

where

$$\zeta_N \;:=\; \frac{1}{2N} \sum_{i \in I_{C_0}} \ln\big(2\ln(1/p_i)\big).$$

The term ζ_N varies only through a log–log dependence on the effective tail probabilities. Because μ, σ, R, and the representative-margin substitution enter only through C_N, we calibrate C_N empirically rather than evaluate this expression (paragraph *Calibration* below).

Comparison to the naive cost. The naive, brute-force method computes N full d-dimensional distances, at cost Nd. The model predicts a cost smaller by the factor α/C_N, i.e. scaling with α up to the slowly varying constant.

On the role of α. The parameter α controls the decay $e^{-\alpha m/d}$. In the fitted exponential model, $\alpha \leq 1$ leaves at least a fraction $e^{-1/2} \approx 0.61$ of the energy after half of the coordinates, so the model predicts weak bounds and little pruning; it predicts substantial pruning only for α well above 1.

Model prediction B.7 (Refinement cost)**.** Let $\phi := \mathbb{E}[\text{Cost}]/(Nd)$ be the average fraction of dimensions processed per candidate. Under Assumptions A1–A4 and the representative margin of Definition B.4, the model predicts

$$\mathbb{E}[\text{Cost}] \;\approx\; \frac{C_N}{\alpha} Nd, \qquad \phi \;\approx\; \frac{C_N}{\alpha}, \qquad C_N \;\approx\; \frac{|I_{C_0}|}{N} \ln \frac{C_0}{\sigma} - \zeta_N,$$

where C_N is the finite-sample constant defined in (4). Equivalently, PANORAMA is predicted to evaluate a fraction C_N/α of the Nd coordinates a brute-force scan would touch, a coordinate-work reduction of α/C_N.

Derivation. Combining the pruning-depth model with the definition of C_N,

$$\phi = \frac{\mathbb{E}[\text{Cost}]}{Nd} \approx \frac{1}{\alpha N} \sum_{i=\max\{i_0,k+1\}}^{N} \min\left\{\alpha, \left[\ln \frac{C_0}{\Delta_i}\right]_+\right\} =: \frac{C_N}{\alpha}.$$

Under the Gaussian margin approximation (5), C_N takes the stated form, so the dependence on N enters only through the averaged log–log term ζ_N. Since the per-candidate work is at most d, failures of $\mathcal{E}_\delta$ change the expectation by at most $\delta N d$: $\mathbb{E}[\text{Cost}] \le \mathbb{E}[\text{Cost} \mid \mathcal{E}_\delta] + \delta N d$, so the prediction does not depend on δ once δ is small. □

Calibration. For a run in which PANORAMA evaluates a measured fraction p of the candidate dimensions , the calibrated constant is $\widehat{C}_N := \alpha p$, with α fitted to the PCA spectrum as in Assumption A1. The semi-empirical speedup of Figure 9 is $1/p = \alpha/\widehat{C}_N$, where p is measured in the same run as the plotted Panorama QPS; it has no other fitted parameter. The calibration is therefore in-sample: each plotted point supplies its own constant. The figure tests the last step of the model, that the saved coordinate work becomes wall-clock speedup, and not the prediction of p from α. Its points use five recall targets spaced evenly over each dataset's Panorama frontier within $[0.8, 1]$; each is paired with the baseline frontier point of nearest recall, and the plotted speedup is their measured QPS ratio. IVFFlat points lie on or below the diagonal, so bookkeeping absorbs part of the saved work and the rest reaches the clock; IVFPQ points spread on both sides by up to $4\times$, so its speedup is not explained by coordinate work alone.

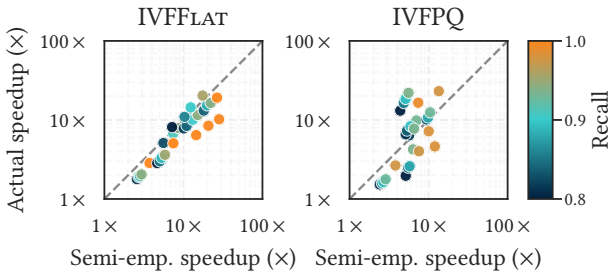


Figure 9: Measured QPS speedup versus the semi-empirical speedup $1/p$, the coordinate-work reduction of the same run, for PANORAMA-IVFFlat and PANORAMA-IVFPQ; color encodes recall.

B.4 THE ε-RELAXATION

Setting $\varepsilon = 1$ uses the exact Cauchy–Schwarz correction $2C^m$ and preserves recall. The full gap $4C^m$ is reached only at the residual alignment $\rho_m = -1$; if residuals are nearly orthogonal ($\rho_m \approx 0$, as Assumption A5 below postulates for non-near-neighbour pairs), the gap is half as large. The ε-relaxation discounts the correction by a factor $\varepsilon \in (0, 1]$, pruning earlier at a recall cost that we measure rather than bound. We record the exact gap identity, state residual isotropy as a modelling assumption, and derive a conditional heuristic prediction that parallels Model prediction B.7.

The ε-relaxed bound is

$$\mathsf{LB}^m_\varepsilon(\mathbf{q}, \mathbf{x}) = A^m - 2\varepsilon\, C^m, \qquad C^m := \sqrt{R^{(m,d)}_{T(\mathbf{q})}\, R^{(m,d)}_{T(\mathbf{x})}},$$

where $A^m = R^{(0,d)}_{T(\mathbf{q})} + R^{(0,d)}_{T(\mathbf{x})} - 2p^{(0,m)}(\mathbf{q}, \mathbf{x})$. Define the residual inner product

$$\pi_m = \sum_{j>m} T(\mathbf{q})_j\, T(\mathbf{x})_j$$

and its normalised value

$$\rho_m := \frac{\pi_m}{C^m} \in [-1, 1],$$

where the bounds follow from Cauchy–Schwarz applied to the residual.

Lemma B.8 (Gap identity)**.** *For every level m,*

$$\|\mathbf{q} - \mathbf{x}\|^2 - \mathsf{LB}^m_\varepsilon(\mathbf{q}, \mathbf{x}) = 2C^m(\varepsilon - \rho_m).$$

Proof. Direct expansion:

$$\|\mathbf{q}-\mathbf{x}\|^2 = R^{(0,d)}_{T(\mathbf{q})} + R^{(0,d)}_{T(\mathbf{x})} - 2\sum_{j=1}^{d} T(\mathbf{q})_j T(\mathbf{x})_j$$
$$= A^m - 2\pi_m.$$

Subtracting the relaxed bound,

$$\|\mathbf{q}-\mathbf{x}\|^2 - \mathsf{LB}^m_\varepsilon(\mathbf{q},\mathbf{x}) \;=\; (A^m - 2\pi_m) - (A^m - 2\varepsilon C^m) \;=\; 2(\varepsilon C^m - \pi_m) \;=\; 2C^m(\varepsilon - \rho_m). \;\square$$

At $\varepsilon = 1$ and worst case $\rho_m = -1$ the gap is $4C^m$, matching the strict cost model; at $\varepsilon = 1$ and $\rho_m \approx 0$ it is $2C^m$. Because $\rho_m \le 1$, $\mathsf{LB}^m_\varepsilon$ can exceed the true distance only when $\rho_m > \varepsilon$, so a true neighbour can be pruned at level m only if $\rho_m > \varepsilon$ there.

Assumption A5 (Residual isotropy)**.** For non-near-neighbour pairs $(\mathbf{q}, \mathbf{x})$, we model the residual directions $(T(\mathbf{q})_j)_{j>m}$ and $(T(\mathbf{x})_j)_{j>m}$, each normalised to unit length, as independent and uniformly distributed on the unit sphere of $\mathbb{R}^{d-m}$. Consequently, by standard high-dimensional concentration,

$$\rho_m \;\sim\; \mathcal{N}\Big(0,\, \tfrac{1}{d-m}\Big) \quad \text{approximately.}$$

Remark. Assumption A5 is an independent modelling assumption about the data, not a consequence of the transform: an orthogonal rotation applied to both vectors, such as the intra-level rotation of Section 3, leaves π_m, C^m, and hence ρ_m unchanged at level boundaries. It is stated only for non-near-neighbour pairs; we make no distributional assumption for true k-NN pairs, which are the only source of recall loss at $\varepsilon < 1$.

Model B.9 (Pruning depth, ε-relaxed)**.** Under Assumptions A1–A5, for non-near-neighbour candidates and a margin $\Delta > 0$, the pruning depth at the ε-relaxed bound is

$$d^{(\varepsilon)}(\Delta) \;\approx\; \frac{d}{\alpha}\ln\frac{2R^2}{\Delta} \;+\; \frac{d}{\alpha}\ln\varepsilon.$$

Derivation. Under A5, ρ_m concentrates at 0 with deviation $1/\sqrt{d-m}$. Substituting $\rho_m \approx 0$ in Lemma B.8,

$$\|\mathbf{q}-\mathbf{x}\|^2 - \mathsf{LB}^m_\varepsilon(\mathbf{q},\mathbf{x}) \;\approx\; 2\varepsilon\, C^m.$$

Applying Assumption A1 to both factors of C^m and Assumption A2,

$$2\varepsilon\, C^m \;\approx\; 2\varepsilon\, \|\mathbf{q}\|\,\|\mathbf{x}\|\, e^{-\alpha m/d} \;\le\; 2\varepsilon\, R^2\, e^{-\alpha m/d}.$$

Pruning occurs at the smallest m for which this is at most Δ:

$$2\varepsilon R^2 e^{-\alpha m/d} \;\le\; \Delta \;\iff\; m \;\ge\; \frac{d}{\alpha}\ln\frac{2\varepsilon R^2}{\Delta} \;=\; \frac{d}{\alpha}\ln\frac{2R^2}{\Delta} \;+\; \frac{d}{\alpha}\ln\varepsilon. \qquad \square$$

The first term is the isotropic-case pruning depth at $\varepsilon = 1$, with prefactor $2R^2$ rather than the strict-mode prefactor $C_0 = 4R^2$. The two prefactors differ by a constant inside the logarithm, which is absorbed into the calibrated constant as in Model prediction B.7.

Heuristic prediction B.10 (Cost under ε-relaxation)**.** Conditional on Assumptions A1–A5 and the representative margin, and for ε with $c_{\text{typ}} + \ln\varepsilon > 0$, the model predicts

$$\mathbb{E}[\text{Cost}_\varepsilon] \;\approx\; \frac{Nd\,(c_{\text{typ}} + \ln\varepsilon)}{\alpha},$$

where c_{typ} is the isotropic-residual counterpart of the constant C_N (with prefactor $2R^2$ instead of $4R^2$). Setting $c_{\text{typ}} = 1$, an assumed value rather than a fitted one, gives $\mathbb{E}[\text{Cost}_\varepsilon] \approx Nd(1+\ln\varepsilon)/\alpha$ for $\varepsilon \in (1/e, 1]$, i.e. $\alpha_{\text{eff}}(\varepsilon) \approx \alpha/(1+\ln\varepsilon)$ and throughput scaling as $1/(1+\ln\varepsilon)$ relative to $\varepsilon = 1$ when refinement dominates query time. The dashed curve of Figure 6 is this scaling, anchored at the measured QPS at $\varepsilon = 1$. In the model, each candidate is pruned $(d/\alpha)\ln(1/\varepsilon)$ dimensions earlier than at $\varepsilon = 1$, until the depth reaches zero.

Derivation. The argument parallels Model prediction B.7, with the pruning depth of Model B.5 replaced by Model B.9, on the same event $\mathcal{E}_\delta$ and with the same representative margins. Substituting (5) into Model B.9 and aggregating as in Model B.6, the additive $(d/\alpha)\ln\varepsilon$ term carries through the sum to give

$$\mathbb{E}[\mathrm{Cost}_\varepsilon] \;\approx\; \frac{Nd}{\alpha}\,(c_{\mathrm{typ}} + \ln\varepsilon),$$

where $c_{\mathrm{typ}} = \ln(2R^2/\sigma) - \zeta_N$ is the isotropic-residual counterpart of C_N (taking $|I_{C_0}| \approx N$ and the cap inactive), shifted only by $\ln 2$ (the isotropic-case prefactor is $2R^2$ rather than $C_0 = 4R^2$) and inheriting the same log–log behaviour. Setting $c_{\mathrm{typ}} = 1$ gives the simplified form. □

Remark (Validity range). The prediction requires $c_{\mathrm{typ}} + \ln\varepsilon > 0$. Setting $c_{\mathrm{typ}} = 1$ gives the threshold $\varepsilon > 1/e \approx 0.37$; the actual threshold is dataset-dependent. Below it, the formula predicts a negative pruning depth—an artifact: the bound clears the margin already at $m = 0$, and the cost is governed by other factors (heap warm-up, fixed per-candidate overhead). In this regime ε should be tuned empirically.

Remark (Additive vs. multiplicative). The ε-relaxation contributes *additively* to the per-candidate pruning depth, in contrast to the multiplicative $1/\alpha$ scaling of the base model. A moderate $\varepsilon = 0.7$ shifts the prefactor by $\ln 0.7 \approx -0.36$, a predicted $1/(1 + \ln 0.7) \approx 1.56\times$ throughput gain at $c_{\mathrm{typ}} = 1$, without changing the leading $1/\alpha$ behaviour; its effect on recall is measured, not predicted.

Heuristic prediction B.10 describes only the speed side of the ε knob. By Lemma B.8, pruning at $\varepsilon < 1$ can discard a true k-nearest neighbour only at a level where its residual ratio ρ_m exceeds ε, so the recall loss depends on the dataset-specific distribution of ρ_m over true k-NN pairs. We neither model nor measure that distribution and give no recall bound; recall at each ε is measured directly (Figure 6).

C IVF–RaBitQ Integration

This appendix details the integration of Section 4 and the RaBitQ curves of Figure 2.

C.1 Sign test and exact-cardinality gate

Both indexes use Faiss's non-FastScan `IndexIVFRaBitQ` (the PANORAMA index is a subclass) with b-bit residual codes, squared-L2 distance, and an uncentered 8-bit query ($q_b = 8$). For a query probed in a list with centroid $\mathbf{c}$, the scorer quantizes the rotated query residual to integers $\bar{q}_1, \ldots, \bar{q}_d \in \{0, \ldots, 255\}$ with binary32 scalars $c_1 \geq 0$, c_2, and c_{34}, and sets $\gamma = \|\mathbf{q} - \mathbf{c}\|^2$ and $g = \sqrt{\gamma}$. A database code stores sign bits $s \in \{0,1\}^d$, $(b-1)$-bit extra codes, the binary32 sign-stage factors $\nu = \|\mathbf{x} - \mathbf{c}\|^2$, $\delta \geq 0$, and $f \geq 0$, and two multi-bit factors. With $A = \sum_j s_j \bar{q}_j$ and $C = \sum_j s_j$, the sign test evaluates, in binary32 with correctly rounded fused multiply-adds,

$$\begin{aligned} F_C(A) &= \mathrm{fl}\big(\mathrm{fma}(c_1, A, \mathrm{fl}(c_2 C)) - c_{34}\big), \\ \ell_C(A) &= \max\big\{0,\ \mathrm{fl}\big(\mathrm{fma}(-2\delta, F_C(A), \mathrm{fl}(\nu + \gamma)) - \mathrm{fl}(fg)\big)\big\}, \end{aligned} \tag{6}$$

and passes the candidate to the multi-bit estimator only if $\ell_C(A) < \tau$, where τ is the current kth-best heap distance.

The gate opens the sign code at the fixed boundary $m = \min(768, d)$. When $m < d$ and the heap is full, it computes the prefix dot $A_{\leq m} = \sum_{j \leq m} s_j \bar{q}_j$, reads the stored counts $C_{\leq m}$ and C, and rejects the candidate if

$$\ell_C\big(A_{\leq m} + T_m(C - C_{\leq m})\big) \;\geq\; \tau, \tag{7}$$

where $T_m(r)$ is the sum of the r largest values among $\bar{q}_{m+1}, \ldots, \bar{q}_d$. Otherwise it reads the remaining $d - m$ sign bits and applies the full sign test to the complete A and the stored C.

Lemma C.1 (Exact-cardinality sign-prefix bound)**.** *Let $0 < m < d$, $r = C - C_{\leq m}$, and $\bar{A} = A_{\leq m} + T_m(r)$. Then $A \leq \bar{A}$, and some sign vector that agrees with s on the first m coordinates and has exactly r ones after them attains $\bar{A}$. If $\sum_j \bar{q}_j \leq 2^{24}$, $c_1 \geq 0$, $\delta \geq 0$, and binary32 arithmetic rounds to nearest, then $\ell_C(\bar{A}) \leq \ell_C(A)$. Hence every candidate rejected by Equation (7) fails the RaBitQ sign test at the same threshold τ.*

Proof. The suffix contributes $\sum_{j>m} s_j \bar{q}_j$, a sum of exactly r of the values $\bar{q}_{m+1}, \ldots, \bar{q}_d$; no r of them sum to more than the r largest, and placing the ones on those coordinates attains $T_m(r)$. Every integer involved is at most $\sum_j \bar{q}_j \le 2^{24}$ and converts to binary32 exactly, so $A \le \bar{A}$ survives the conversion. Round-to-nearest is nondecreasing, so $\mathrm{fma}(c_1, \cdot, \mathrm{fl}(c_2 C))$ with $c_1 \ge 0$ and the subtraction of c_{34} are nondecreasing in A, $\mathrm{fma}(-2\delta, \cdot, \mathrm{fl}(\nu + \gamma))$ with $-2\delta \le 0$ is nonincreasing, and subtracting the A-independent $\mathrm{fl}(fg)$ and clamping at zero preserve order. Thus ℓ_C is nonincreasing in A, and $\ell_C(\bar{A}) \ge \tau$ implies $\ell_C(A) \ge \tau$, which is the RaBitQ rejection condition. □

The stored counts make the bound exact in cardinality: given $C_{\le m}$ and C, $T_m(r)$ is the largest suffix contribution of any sign pattern consistent with the sidecar. Since Equation (7) evaluates the scorer's own binary32 expression itself, it needs no rounding margin. A rejected candidate is one the RaBitQ scorer also discards before multi-bit scoring, so it cannot enter the result heap under either scorer. The gate has no relaxation parameter; in this strict mode, its only one and the analogue of $\varepsilon = 1$, it compares against the unmodified heap threshold, and any failed guard (a non-finite or negative factor, a count outside its range, or a query violating the conditions of Lemma C.1) sends the candidate to the RaBitQ sign test. The table $T_m(0), \ldots, T_m(d-m)$ ($d - m + 1$ `uint32` entries, 772 bytes for GIST) is built once per probed list, since the quantized residual depends on the list, from a 256-bin histogram of $\bar{q}_{m+1}, \ldots, \bar{q}_d$.

C.2 Boundaries, sidecar layout, and survivor scoring

The sign code is read over the deduplicated fixed boundaries $\{0, \min(768, d), d\}$, which gives the ranges in Table 3, for example $96 + 24$ sign bytes for GIST's $768 + 192$ split. For Deep, SIFT, and arXiv ($d \le 768$) the schedule is a single range, so no prefix test is evaluated and the gate reduces to the RaBitQ sign test computed from the stored count C. Each byte-aligned range is scored by a dot-only AVX2 AND-popcount kernel, and the range dots sum exactly to A.

Each inverted-list row is the RaBitQ code, byte for byte, followed by an eight-byte sidecar. The RaBitQ code holds $\lceil d/8 \rceil$ sign bytes, 12 bytes for (ν, δ, f), $\lceil d(b-1)/8 \rceil$ extra-bit bytes, and 8 bytes of multi-bit factors, so it occupies $D = 92$, 116, 692, 860, 1364, and 2708 bytes on Deep, SIFT, arXiv, GIST, Ada, and Large. At offsets D, $D + 2$, and $D + 4$ the sidecar stores the cumulative sign popcounts $C_{\le \min(512,d)}$ (`uint16`), $C_{\le \min(768,d)}$ (`uint16`), and $C = C_{\le d}$ (`uint32`), all written at encoding time. Search reads only the last two fields; the 512-dimension count is stored and validated but not read. The sidecar is the only persistent difference between the indexes: exactly 8 bytes per vector (Table 2), which raises the logical persistent index size (codes, IDs, centroids, and rotation) by between 0.29% (Large) and 8.0% (Deep).

A candidate that passes the sign test is scored from the untouched RaBitQ code bytes by the unchanged multi-bit estimator, and the sidecar never enters a score. For 7-bit codes the scanner runs the RaBitQ scorer's AVX2 fused-multiply-add accumulation and shared finalizer inline; for 6-bit codes (Deep and SIFT) it calls the RaBitQ distance computer.

Table 3: IVF–RaBitQ configuration per dataset. Sign ranges are the dimensions opened before and after the fixed boundary. Rejected is the share of scanned candidates rejected by Equation (7), and Sign B/cand. the mean sign bytes read per candidate by the PANORAMA index, both as ranges over the nprobe grid. "–" marks datasets without a prefix boundary.

Dataset	d	nlist	b	Sign ranges	nprobe values	Rejected (%)	Sign B/cand.
Deep	96	1024	6	96	19	–	12 / 12
SIFT	128	1024	6	128	17	–	16 / 16
arXiv	768	256	7	768	17	–	96 / 96
GIST	960	128	7	768+192	16	82.4–99.3	96.2–100.2 / 120
Ada	1536	128	7	768+768	16	4.0–49.5	144.5–188.2 / 192
Large	3072	128	7	768+2304	16	0.09–0.77	381.8–383.7 / 384

C.3 Proof of Proposition 4.1 and identity checks

The proposition follows from Lemma C.1 by induction over the scan, and the checks below confirm it on every measured setting.

Proof of Proposition 4.1. The two indexes use bit-identical rotation matrices, coarse centroids, and list assignments, and every row begins with the plain code, so for a fixed query both scan the same candidates in the same order with the same quantized query. We induct over this sequence with the result heap as invariant: assume both heaps agree before candidate $\mathbf{x}$, so τ agrees. If the gate rejects $\mathbf{x}$ at the boundary, Lemma C.1 shows that the RaBitQ sign test rejects it as well. Otherwise the complete A is a sum of exact integer range dots, the stored C equals the popcount of the full code, and Equation (6) is evaluated with the same operations in the same order, so the sign-test outcome is reproduced bit for bit. A survivor's multi-bit score is computed from identical bytes by the identical instruction sequence, so the same (distance, ID) pair reaches the heap; every guard failure routes to the RaBitQ functions themselves. Hence the heaps agree after every candidate, and the returned IDs and distances are bitwise identical. □

Identity checks. For each dataset and each of the 101 nprobe settings in Table 3, all 1000 queries were searched by three configurations: plain IVF–RaBitQ, the PANORAMA index with the prefix test disabled, and the PANORAMA index with the gate. The harness compared top-10 IDs exactly and float32 distances as bit patterns for every pair of configurations, checked that all three made the same aggregate numbers of sign-test refinements and rejections, and all 101 settings matched. Before search it verified bit-identical rotation matrices and coarse centroids, identical list assignments, a byte-identical RaBitQ prefix in every row, and every stored count against a recomputation.

C.4 EXPERIMENTAL PARAMETERS

Both arms place a Faiss `RandomRotationMatrix` (seed 123) in front of the IVF index through `IndexPreTransform`; the PANORAMA index receives a separately constructed, bit-identical rotation and a copy of the plain index's `IndexFlatL2` coarse centroids. Both are trained on and filled with the complete base set (32 build threads), and Faiss is compiled with AVX2. The nprobe grid is $\{1, 2, 4, 6, 8, 12, 16, 20, 24, 28, 32, 40, 52, 64, 88, 128\}$ for GIST, Ada, and Large, extended by 256 for arXiv and by 256, 512, and 1024 for Deep; SIFT uses $\{1, 2, 3, 4, 6, 8, 12, 16, 24, 32, 48, 64, 96, 128, 256, 512, 1024\}$. Each point searches the first 1000 rows of the query file with $k = 10$ and reports recall@10 against exact ground truth. Search runs on one pinned core with all OpenMP and BLAS thread counts set to one, and the query rotation is inside the timed batched search call. Each setting is timed three times with the two arms alternating in cyclic order, and QPS is the mean over repetitions. The recall-0.95 ratios in Section 5 interpolate each arm's QPS linearly in recall within its measured range.

D RECALL COVERAGE AND PARAMETERS

Table 4 lists which datasets each comparison covers. The frontier figures of Section 5 display recall@10 between 0.8 and 1.0. Table 6 gives the full measured range of every plotted frontier, so a curve that stops inside the window can be read as ending at its last measurement. Three cases matter for the comparisons. KS2 on arXiv reaches at most recall 0.8744 on its tested grid ($L = 48$, $efSearch \leq 1024$), so no matched-recall comparison with KS2 is made above it. VAQ's memory-matched tier spans a narrow interval on each dataset (for example 0.88–0.89 on Deep and 0.79–0.87 on GIST), and its curve in Figure 2 covers only that interval. Figure 12 shows that VAQ's $2\times$ tier reaches higher recall than its memory-matched tier on every dataset. In the authors-code graph comparison, DADE includes $efSearch$ values from 10 to 80 on the same graphs as its $efSearch \geq 100$ points (on Large also 12 and 16), so its curves cover the low end of the window. Figure 11 repeats that comparison over the wider window 0.6–1.0; a few endpoints still lie to its left (GIST: Panorama 0.48, DADE 0.42, hnswlib 0.34; Deep: hnswlib 0.59).

Table 4: Dataset coverage of each comparison.

Comparison	Coverage	Note
IVF-Flat: Faiss, PDX, ADSampling, DADE, DDC	6/6	All methods scan full-precision vectors.
Compressed IVF: Faiss-PQ, Panorama-PQ, RaBitQ pair, VAQ	6/6	VAQ: compressed-only, memory-matched.
Faiss-family HNSW	6/6	Shared graph except ADSampling.
Authors-code HNSW: Panorama, DADE, hnswlib	6/6	Each system builds its own graph.
FastScan refinement pair	6/6	Exact full-precision reranking.
Transform, systems, fixed-graph ablations	GIST	Single-dataset scope.

Table 5 lists the per-dataset parameters of the frontier experiments; *nprobe* and *efSearch* are swept, and all PQ codes use 8-bit subquantizers. DADE uses its default $P_s = 0.1$ and $\Delta d = 32$, and VAQ's tiers follow a memory-matching rule of d and $2d$ code bytes.

Table 5: Per-dataset parameters. L is Panorama's level count, ε its relaxation, and κ the IVFPQ shaping cap; M_{PQ} is the number of 8-bit PQ subquantizers (bytes per code); IVFPQ uses the IVF-Flat nlist, and every Faiss-family HNSW graph uses $M = 32$.

	IVF-Flat			IVFPQ					Faiss-family HNSW		
Dataset	nlist	L	ε	M_{PQ} Faiss	M_{PQ} Pano.	L	κ	ε	L	ε	KS2 L
Deep	1024	4	1.0	96	96	4	4	1.0	2	1.0	6
SIFT	1024	8	1.0	32	128	8	8	0.9	2	1.0	8
arXiv	256	24	0.7	768	768	16	3	0.8	8	0.7	48
GIST	128	30	0.5	160	960	16	8	0.8	8	0.5	60
Ada	128	16	0.8	1536	1536	16	4	0.7	8	0.7	96
Large	128	32	0.7	1024	3072	32	3	0.8	8	0.7	192

Figure 6 sweeps the four runtime knobs of PANORAMA on Ada with every inverted list probed, so that the candidate stream is fixed and only the verification changes; Section 5 summarizes it.

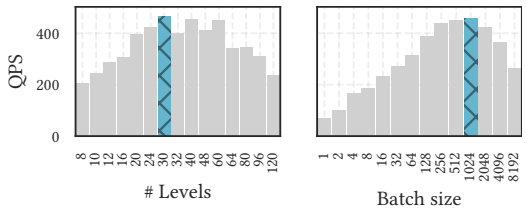


Figure 10: Layout knobs on Ada: IVF-Flat QPS versus the number of levels and the batch size; the selected setting is highlighted.

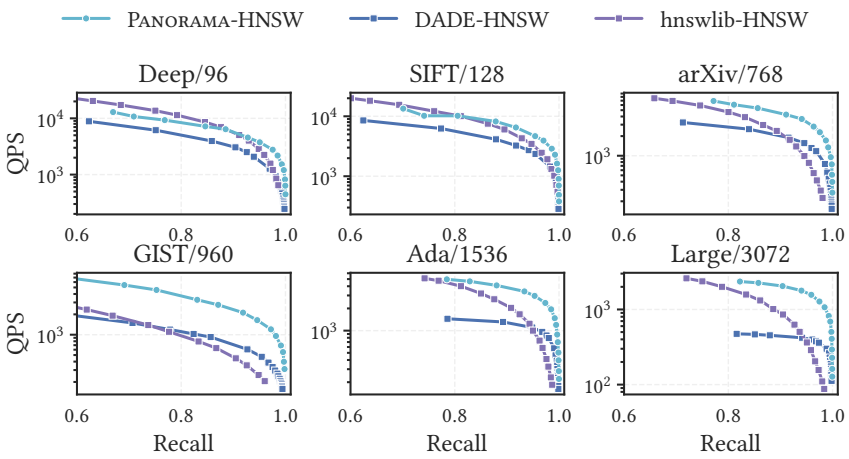


Figure 11: Authors-code graph comparison of Figure 4 over the wider window recall@10 0.6–1.0.

Table 6: Measured recall@10 range (minimum–maximum) of each plotted Pareto frontier, rounded to two decimals; "–" marks a configuration that was not run. The main-text figures display the window 0.8–1.0.

Method	Deep	SIFT	arXiv	GIST	Ada	Large
IVF-Flat (Figure 1)						
Panorama-IVFFlat	0.51–1.00	0.47–1.00	0.50–1.00	0.40–1.00	0.70–1.00	0.64–1.00
Faiss-IVFFlat	0.51–1.00	0.47–1.00	0.50–1.00	0.40–1.00	0.70–1.00	0.64–1.00
PDX-IVFFlat	0.48–1.00	0.44–1.00	0.45–1.00	0.37–0.99	0.67–1.00	0.61–1.00
ADSampling-IVFFlat	0.51–1.00	0.47–1.00	0.50–1.00	0.40–1.00	0.70–1.00	0.64–1.00
DADE-IVFFlat	0.51–1.00	0.47–1.00	0.50–1.00	0.40–1.00	0.70–1.00	0.64–1.00
DDC-IVFFlat	0.51–1.00	0.47–1.00	0.50–1.00	0.39–0.99	0.69–0.98	0.63–0.99
Compressed IVF (Figures 2 and 12)						
Panorama-IVFPQ	0.51–0.98	0.47–0.98	0.50–0.98	0.40–0.98	0.69–0.98	0.64–0.99
Faiss-IVFPQ	0.51–0.98	0.47–0.96	0.50–0.99	0.40–0.98	0.70–0.99	0.64–0.99
Panorama-IVF-RaBitQ	0.50–0.97	0.47–0.97	0.50–0.99	0.40–0.99	0.70–0.99	0.64–1.00
IVF-RaBitQ	0.50–0.97	0.47–0.97	0.50–0.99	0.40–0.99	0.70–0.99	0.64–1.00
VAQ, memory-matched	0.88–0.89	0.88–0.89	0.86–0.95	0.79–0.87	0.92–0.95	0.89–0.96
VAQ, 2× code (Fig. 12)	0.94–0.96	0.95–0.96	0.88–0.97	0.84–0.95	0.93–0.97	0.90–0.98
Faiss-family HNSW (Figure 3)						
Panorama-HNSW (Faiss)	0.27–1.00	0.26–1.00	0.30–0.99	0.12–0.99	0.41–0.99	0.40–0.99
Faiss-HNSW	0.27–1.00	0.27–1.00	0.32–0.99	0.13–0.99	0.40–0.99	0.39–0.99
ADSampling-HNSW	0.59–0.99	0.60–1.00	0.66–0.97	0.34–0.95	0.74–0.98	0.72–0.98
KS2-HNSW	0.42–0.99	0.59–1.00	0.07–0.87	0.40–0.97	0.75–0.99	0.76–0.99
Authors-code HNSW (Figure 4)						
Panorama-HNSW (hnswlib)	0.67–1.00	0.70–1.00	0.77–1.00	0.48–1.00	0.78–1.00	0.82–1.00
DADE-HNSW	0.62–1.00	0.62–1.00	0.71–1.00	0.42–0.99	0.79–1.00	0.82–1.00
hnswlib-HNSW	0.59–0.99	0.60–1.00	0.66–0.98	0.34–0.96	0.74–0.99	0.72–0.98
Exact reranking (Figure 5)						
Panorama-FastScan-Refine	0.51–0.99	0.47–1.00	0.50–1.00	0.40–1.00	0.70–1.00	0.64–1.00
Faiss-FastScan-Refine	0.51–0.99	0.47–1.00	0.50–1.00	0.40–1.00	0.70–1.00	0.64–1.00

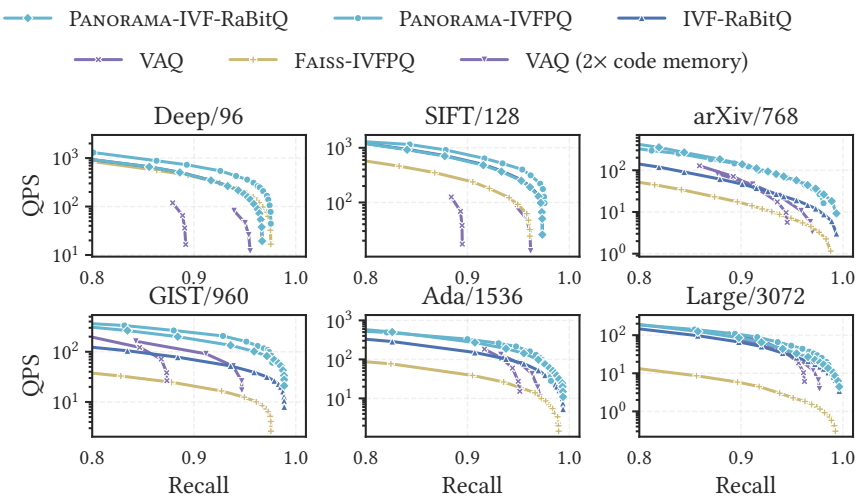


Figure 12: VAQ sensitivity to compressed code budget. The comparison in Figure 2 uses the memory-matched tier.