

# Hierarchical Trajectory Routing in the Transform Domain for Efficient Approximate Nearest Neighbour Search

Jonathan H. West

## Abstract

Distributed Transform-Domain Representation (DTDR) stores numerical data directly in a globally distributed transform basis. Because similarity information is spectrally distributed rather than locally concentrated, DTDR enables hierarchical routing strategies that operate on aggregated transform-domain structure.

This paper demonstrates that a hierarchical trajectory routing mechanism integrated into an IVF-based approximate nearest neighbour (ANN) pipeline achieves equivalent recall to flat IVF while evaluating substantially fewer candidate vectors. On the full SIFT1M benchmark (1M vectors, 10,000 queries), the proposed method achieves recall@10 of 0.580 at nprobe=8 while evaluating approximately 899 candidates per query, compared to approximately 7,812 candidates for flat IVF1024 at the same probe count.

The results demonstrate that transform-domain structure can be leveraged to provide principled within-list routing, yielding significant computational reduction without loss of recall.

## 1 Introduction

Approximate nearest neighbour (ANN) search is a core primitive in modern machine learning systems. Common index structures such as IVF and HNSW partition the search space to reduce candidate evaluations. In standard IVF, each probed list is exhaustively scanned.

Distributed Transform-Domain Representation (DTDR) stores vectors in a globally distributed basis. Similarity information becomes spectrally decomposed across transform coefficients. This structure enables hierarchical aggregation and routing strategies that are not available in conventional coordinate representations.

This paper introduces a hierarchical trajectory routing mechanism that operates within IVF lists using transform-domain aggregation, reducing the number of candidate vectors evaluated while maintaining recall.

## 2 Transform-Domain Similarity Structure

Let  $x_i \in \mathbb{R}^d$  denote database vectors and  $q$  a query vector.

Under an orthogonal transform  $H$ :

$$\tilde{x}_i = Hx_i, \quad \tilde{q} = Hq$$

Similarity becomes:

$$q \cdot x_i = \tilde{q} \cdot \tilde{x}_i$$

Because  $H$  is orthogonal, energy is preserved, but similarity information is redistributed across coefficients.

Aggregating groups of vectors in the transform domain produces structured intermediate representations whose similarity to  $q$  can be evaluated with a single dot product.

### 3 Hierarchical Trajectory Routing

Within each IVF list, vectors are grouped into fixed-size bags (32 vectors per bag in experiments). For each bag, a binary tree of segment means is precomputed.

Each node stores:

$$\mu = \frac{1}{k} \sum_{i=1}^k x_i$$

and its squared norm  $\|\mu\|^2$ .

At query time, node scoring uses an L2 proxy:

$$\text{score} = 2q \cdot \mu - \|\mu\|^2$$

Because node norms are precomputed, each node evaluation requires only one dot product.

Beam descent through the tree localises promising leaf groups. Only leaf candidates are passed to exact L2 reranking.

### 4 Integration with IVF

The full pipeline is:

1. IVF coarse routing over  $nprobe$  centroid lists.
2. Within each list, bag-level scoring.
3. Beam descent over hierarchical tree nodes.
4. Exact L2 reranking over leaf candidates.

Flat IVF evaluates all vectors in each probed list. The trajectory router evaluates only selected leaf groups.

### 5 Experimental Setup

Dataset: SIFT1M (1,000,000 vectors, 128D)

Index:

- IVF1024
- Bag size = 32
- Beam width = 2

Evaluation:

- 10,000 query vectors
- Standard recall@10
- Unfiltered ground truth

## 6 Results

<i>nprobe</i>	recall@10	Mean Candidates	Flat IVF Candidates
8	0.580	899	~7,812
16	0.657	1,796	~15,625
32	0.698	3,577	~31,250

Table 1: Full SIFT1M results (10,000 queries)

At  $nprobe = 8$ , recall matches the published IVF1024 flat baseline while evaluating approximately  $8.7\times$  fewer candidates.

Candidate counts scale approximately as:

$$\text{candidates} \approx nprobe \times 112$$

providing predictable computational behaviour.

## 7 Discussion

The trajectory router does not widen the search arbitrarily. Increasing the number of selected bags improves recall without increasing candidate counts, demonstrating discriminative routing rather than brute-force expansion.

This method complements IVF coarse routing. It does not replace graph-based methods such as HNSW but provides structured within-list localisation using transform-domain aggregation.

## Code Availability

The full implementation of the hierarchical trajectory routing method, including the complete SIFT1M (1M vector) experiments and evaluation scripts used in this paper, is publicly available in the DTDR repository:

[https://github.com/UnrealJon/DTDR/tree/main/experiments/02\\_dtdr\\_end\\_to\\_end\\_search](https://github.com/UnrealJon/DTDR/tree/main/experiments/02_dtdr_end_to_end_search)

The repository contains the notebooks and scripts corresponding to the IVF1024 trajectory routing experiments reported here, including candidate-count measurements and full 10,000-query recall evaluations.

## Intellectual Property Notice

Distributed Transform-Domain Representation (DTDR) and related hierarchical trajectory routing methods described in this paper are the subject of a UK patent application under accelerated examination (Green Channel), UK Patent Application No. GB2602157.6.

The code repository is provided for research and evaluation purposes. Commercial use may require appropriate licensing.

## 8 Conclusion

We have demonstrated that DTDR-compatible hierarchical routing can be integrated into an IVF ANN pipeline to achieve equivalent recall with substantially fewer candidate evaluations.

The results show that transform-domain structure enables principled trajectory navigation within ANN indices, yielding measurable computational efficiency at benchmark scale.