**Distributed Kd-Trees for Retrieval from Very Large Image Collections**

**Goal**
Scaling image retrieval to $10^9$ images:
- Efficient Parallel Implementation
- Measuring Retrieval Precision
- Measuring Retrieval Speed

**Background**
Full Representation (FR) has much better precision than Bag of Words (BoW)

**Independent Kd-Trees (IKdt)**
- Dataset partitioned into M subsets
- Nearest Neighbor search runs in parallel
- Root machine selects nearest of M results

**Distributed Kd-Trees (DKdt)**
- One global Kd-Tree is built from all data
- Top subtree stored in root machine, bottom in leaf machines
- Root machine directs queries to subset of leaf machines

**Results**
- DKdt is clearly superior to IKdt with precision that is 32% higher at 100M images.
- DKdt is superior to IKdt in terms of both precision and throughput. It is about 30 times faster at 100M images.

**Conclusions**
- Implemented and tested parallel retrieval architecture
- Experiments on $10^8$ images and 2000 CPUs
- Scales well with dataset size:
  - Speed unchanged
  - Gentle performance loss

**Reference**