Similarity Search for Large-Scale Feature-Rich Data

Qin (Christine) Lv
Stony Brook University

Joint work with Kai Li, Moses Charikar,
Zhe Wang, William Josephson, Wei Dong
Princeton University

Massive Amounts of Data

- "How much information", 2003 *[Leman & Varian]*
  - 5 exabytes (10^{18}) of new information in 2002
  - 92% stored on magnetic media, mostly hard disks
  - 30% growth per year between 1999 and 2002
- Fast increase of non-text data
  - Audio, video, digital photos, scientific data, …

Flickr	30% growth / month
YouTube	65,000 videos uploaded / day
CERN LHC	1.8 GB / second

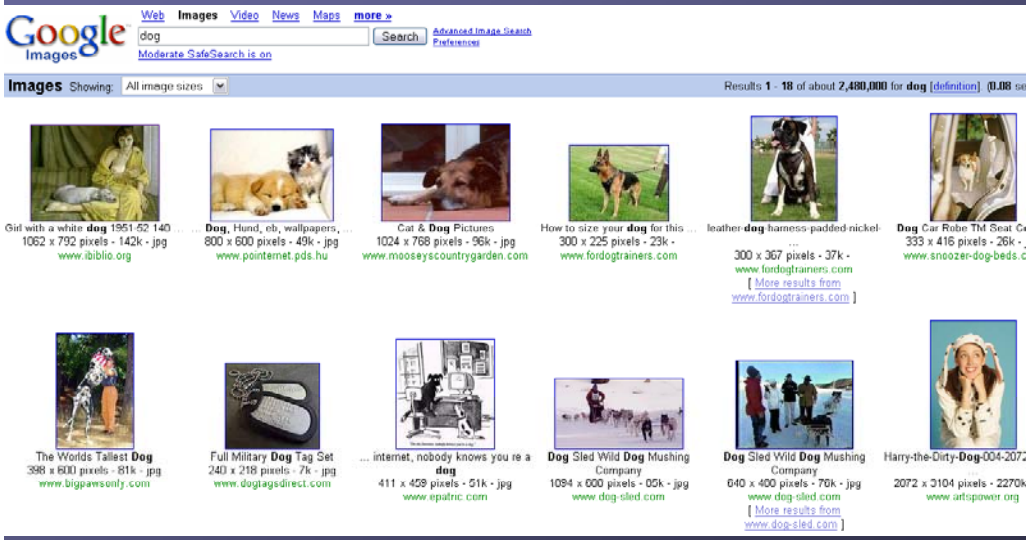
Feature-Rich Data

- Rich amounts of information
- Not easily captured by words
- Fuzzy in nature

A picture is worth a thousand words.

Managing and Searching Feature-Rich Data

- Text-based search techniques inadequate
  - Search on filenames or text annotations
  - Manual annotation is difficult for large datasets
  - Annotations are not perfect



Content-Based Similarity Search

COREL dataset (60,000 images)

Managing and Searching Feature-Rich Data

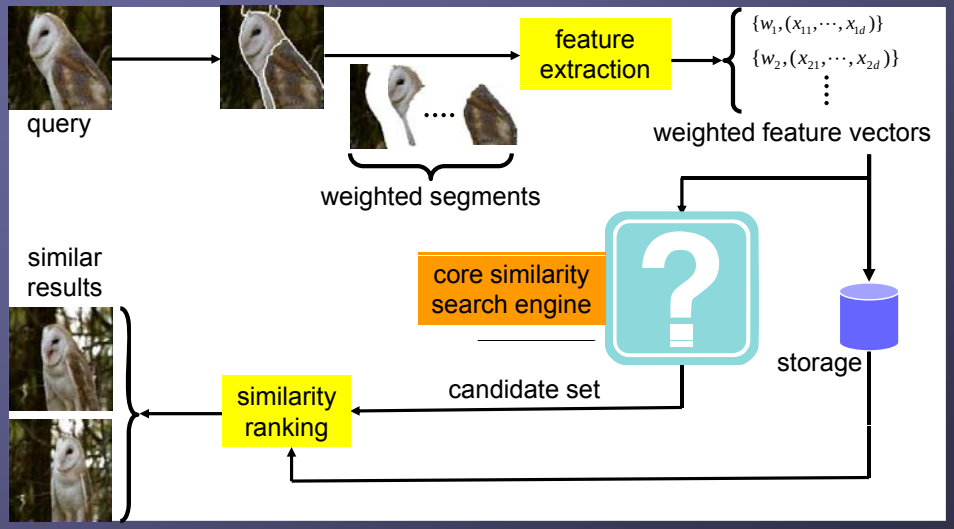
- Text-based search techniques inadequate
  - Search on filenames or text annotations
  - Manual annotation is difficult for large datasets
  - Annotations are not perfect
- Current content-based similarity search
  - Domain efforts focus on feature extraction and similarity measure
  - Limited to small datasets

~~Build efficient similarity search systems~~ for large-scale feature-rich data

Similarity Search Problem

- Given a query, find objects with similar features
  - e.g. color, shape, pose, …
  - Domain-specific
  - Subjective
- Mathematical definition
  - (set of) feature vectors
    - $X=(x_1,x_2, \dots , x_d)$
    - $Y=(y_1,y_2, \dots, y_d)$
    - e.g. color histogram
  - Distance measure D(X,Y)
    - e.g. Euclidean distance, Manhattan distance, …

Similarity Query Processing



Main Challenges & Contributions

- Large metadata size
  - Sketch construction for compact representation
  - Order of magnitude space reduction
- Indexing for high-dimensional similarity search
  - Multi-probe LSH indexing
  - High-efficiency, order of magnitude space reduction
- Complex distance measure
  - Multi-feature filtering for fast query processing
  - Order of magnitude speedup
- Ferret toolkit
  - General-purpose, easy construction, 6 systems built

Outline

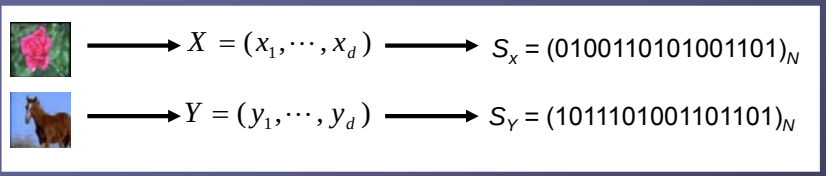
- Motivations
- Sketch construction
  - compact metadata representation
- Multi-probe LSH indexing
  - high-dimensional similarity search
- Ferret toolkit
- Conclusions & future work

Sketch

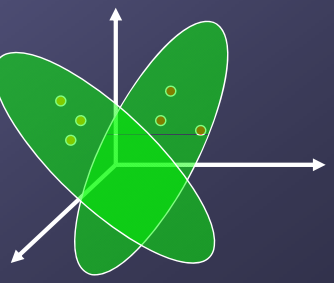
- Compact data representation
- Estimates certain properties of original data

- Previous work in the theory community
  - Set membership *[Bloom'70]*
  - Set similarity *[BGMZ'97, BCMF'98]*
  - Frequency moments *[AMS'99]*
  - String edit distance *[Batu et al. '03]*
  - EMD on points in Euclidean space *[Indyk & Thaper '03]*
  - …

Sketch for Similarity Search



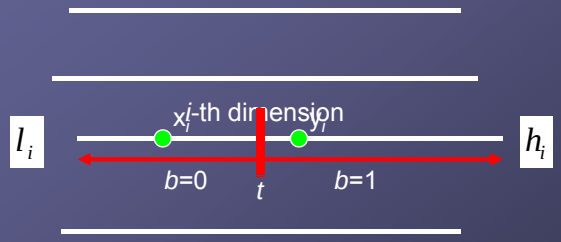
- High-dim feature vec ➔compact bit vec
- Sketch distance approximates feature vector distance



[Charikar, STOC'02]

Sketch Construction

- *d*-dimensional feature vector → *BxH* bit vector
  - Mapping L_1 distance to Hamming distance



$$X = (x_1, x_2, \ldots, x_d)$$
$$Y = (y_1, y_2, \ldots, y_d)$$

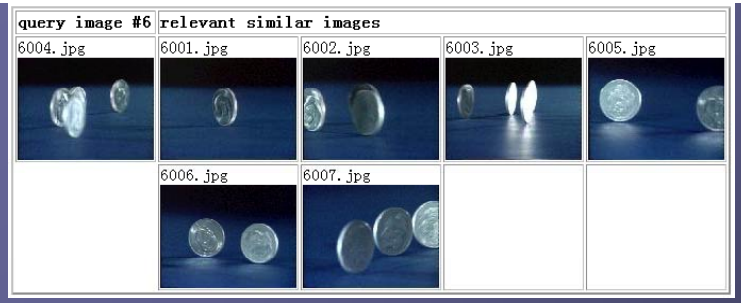
$$\Pr[b_x \neq b_y] = \Pr[t \text{ falls within } x_i \text{ and } y_i] = \frac{|x_i - y_i|}{h_i - l_i}$$

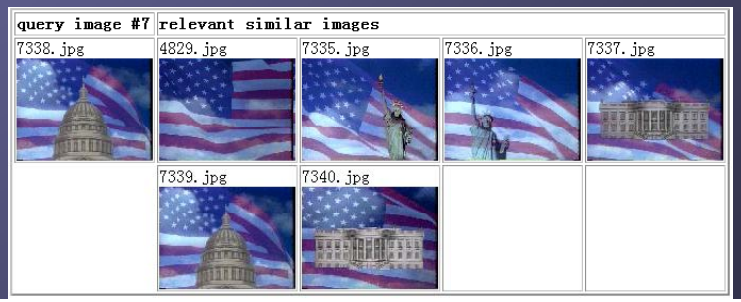
Performance Evaluation

- Tradeoff between search quality and sketch size?
- How much space savings using sketches?

Evaluation Methodology

- Benchmarks
  - VARY image: 10,000 images, 14-d, 32 sets
  - TIMIT audio: 6,300 sentences, 192-d, 450 sets
  - PSB shape: 1,814 3D models, 544-d, 92 sets





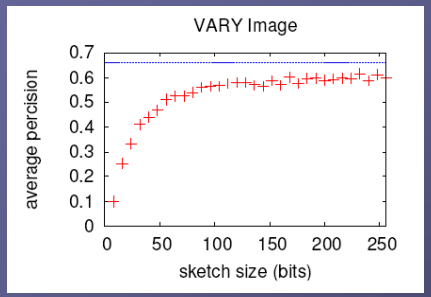
Evaluation Methodology

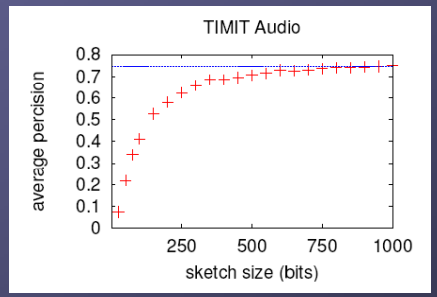
- Search quality measures
  - S: similarity set
  - R: retrieved results
  - Precision: | S ∩ R | / | R |
  - Recall: | S ∩ R | / | S |

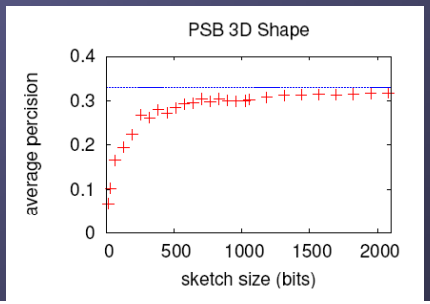
$$average\ precision = \frac{1}{|S|}\sum_{i=1}^{|S|}\frac{i}{rank_i}$$

  - e.g. S = { a, c, g }
    - R1 = { a, b, c, d, e, f }
    - R2 = { d, e, b, a, f, c}

Search Quality vs. Sketch Size







Space Savings using Sketches

Dataset	#Dimensions	Feature vec size (bits)	Sketch size (bits)	Space savings
VARY image	14	448	96	4.7x
TIMIT audio	192	6,144	600	10.2x
3D shape model	544	17,408	800	21.8x

Order of magnitude space reduction using sketches

Outline

- Motivations
- Sketch construction
  - compact metadata representation
- Multi-probe LSH indexing
  - high-dimensional similarity search
- Ferret toolkit
- Conclusions & future work

Indexing for Similarity Search

- Traditional indexing techniques
  - R-tree *[Gut'84]*, R*-tree *[BKSS'90]*, X-tree *[BKK'96]*, SR-tree *[KS'97]*, …
  - "curse of dimensionality"
  - Linear scan outperforms when d > 10 *[WSB'98]*
- Indexing for high-dimensional similarity search

Image SIFT local features	128 dimensions
Audio MFCC features	192 dimensions
3D Shape SHD features	544 dimensions

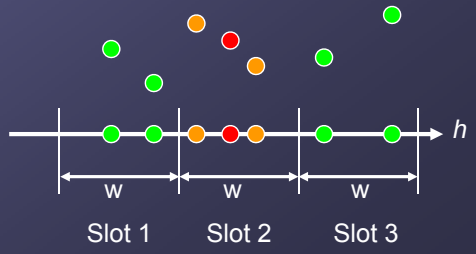
LSH: Locality Sensitive Hashing

- *[Indyk & Motwani, STOC'98]*
- Locality sensitive hashing

$$\Pr[h(u) = h(v)] \text{ is strictly decreasing in } d(u,v)$$

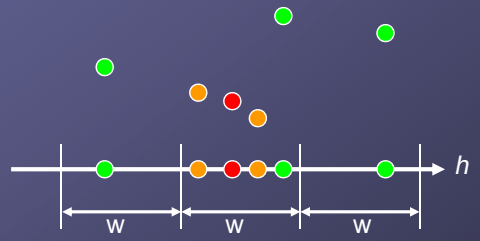
  - *i.e.* closer objects have higher collision probability
- LSH for Euclidean distance
  - *w* : slot width

$$h_{a,b}(v) = \left\lfloor \frac{a \cdot v + b}{w} \right\rfloor$$

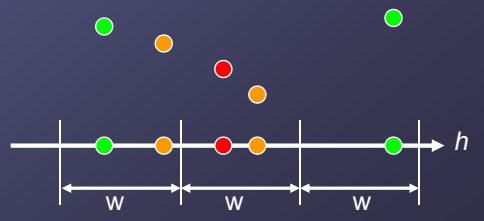


LSH: Main Issues

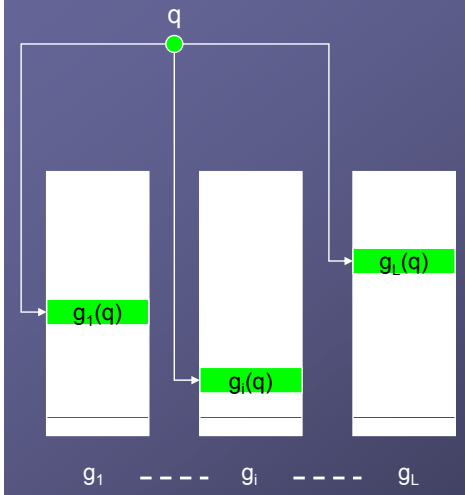
- False positive
  - Far away objects fall into same slot



- False negative
  - Similar objects fall into different slot

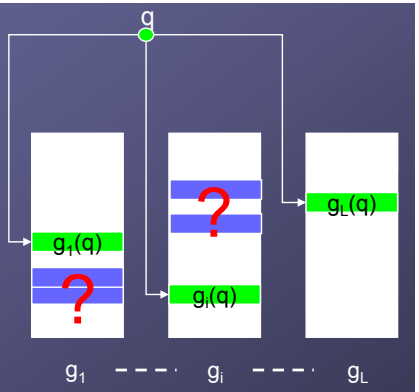


Basic LSH Indexing



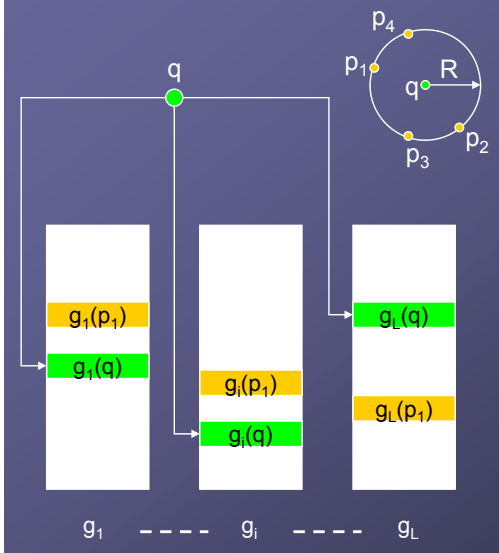
- *[IM98, GIM99, DIIM04]*
- *M* hash functions/table
  $g_i(v) = (h_{i,1}(v), \ldots, h_{i,M}(v))$
- *L* hash tables
  $G = \{ g_1, \ldots, g_L \}$
- Issues:
  - Large number of tables
  - L > 100 in *[GIM99]*
  - L > 500 in *[Buhler01]*

Impractical for large datasets



- Goal
  - Use fewer hash tables
- Approach
  - Probe multiple buckets in each hash table
- Which buckets should we probe?

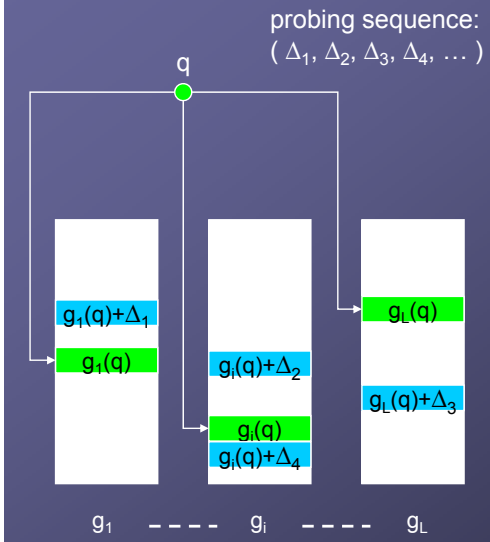
Entropy-Based LSH Indexing



- *[Panigrahy, SODA'06]*
- Randomly perturb *q* at distance *R*
- Check hash buckets of perturbed points
- Issues:
  - Difficult to choose *R*
  - Duplicate buckets

Inefficient probing

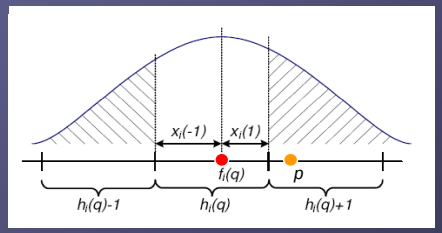
Multi-Probe LSH Indexing



- A carefully derived probing sequence
- Probe directly on hash values
- Pros
  - Fast probing sequence generation
  - No duplicate buckets
  - More effective in finding similar objects

Query-Directed Probing

- Hashed position within slot matters!
  - If q is closer to the right boundary
    - Similar objects more likely in the right slot than in the left slot

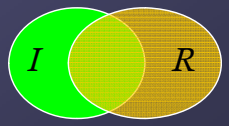


- Query-directed probing
  - Estimate success probability based on $x_i(-1)$ and $x_i(1)$
  - Probe buckets that have high success probability

Evaluation Methodology

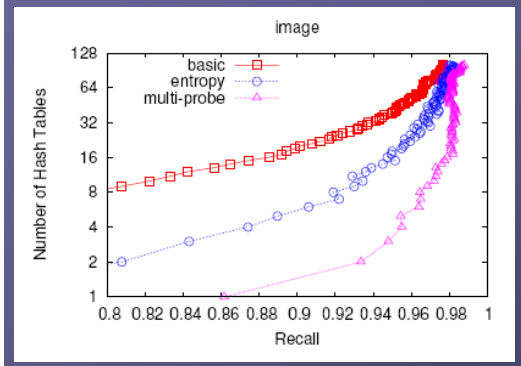
Dataset	#objects	#dimensions
Web images	1.3 million	64
Switchboard audio	2.6 million	192

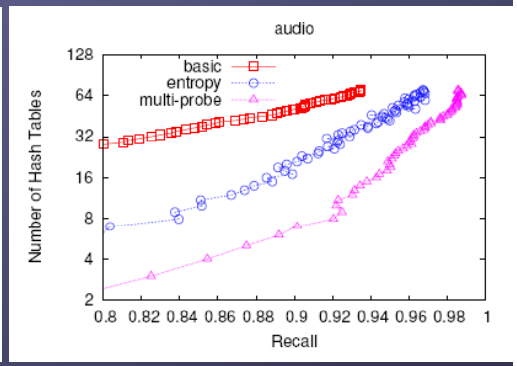
- Benchmarks
  - 100 random queries, top K results
- Evaluation metrics
  - Search quality: recall
  - Search speed: query latency
  - Space usage: #hash tables



$$\text{recall} = |I \cap R| \ / \ |I|$$

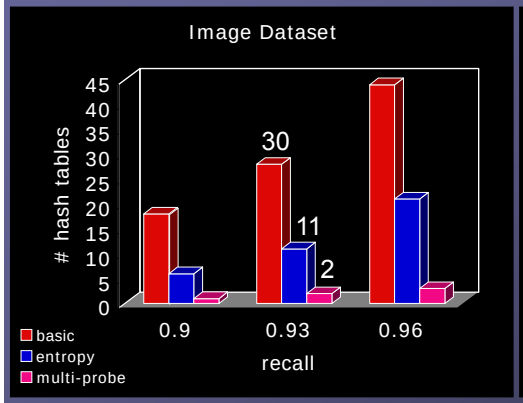
Multi-Probe vs. Basic vs. Entropy

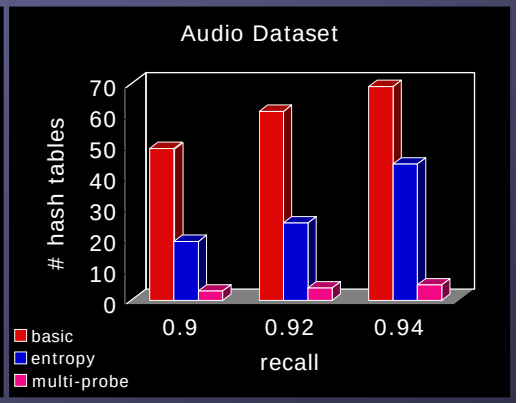




Multi-probe LSH achieves higher recall with fewer hash tables

Space Savings of Multi-Probe LSH





14x - 18x fewer tables than basic LSH
5x - 8x fewer tables than entropy LSH

Outline

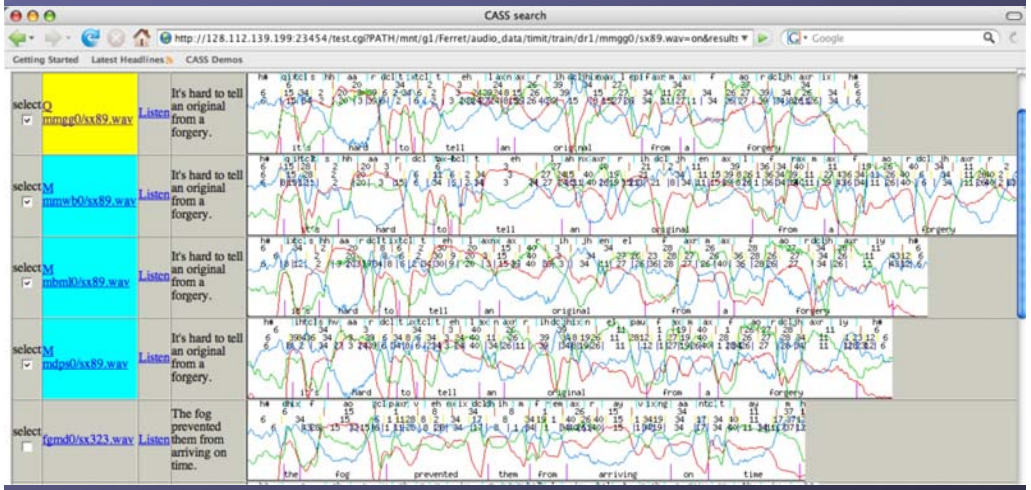
- Motivations
- Sketch construction
  - compact metadata representation
- Multi-probe LSH indexing
  - high-dimensional similarity search
- Ferret toolkit
- Conclusions & future work

Ferret Toolkit

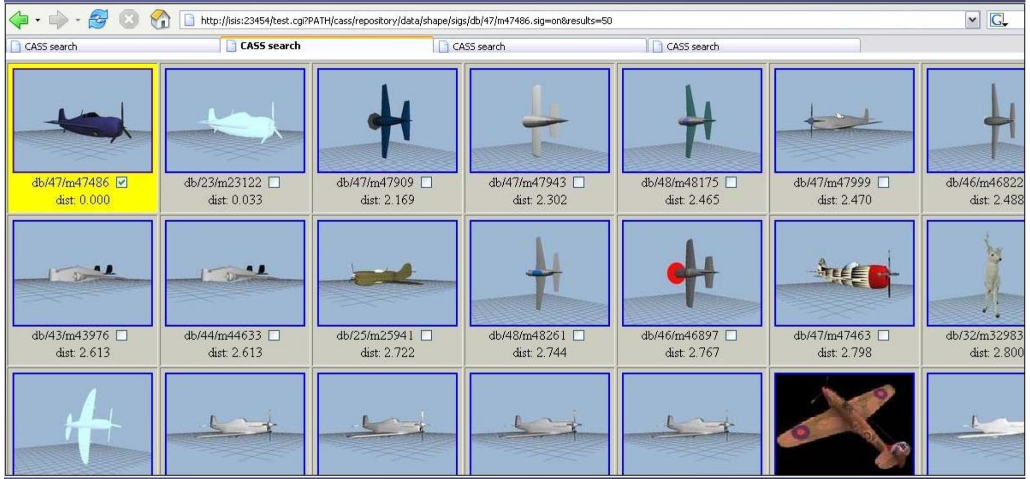
- General-purpose toolkit for building efficient similarity search systems
- Plug-ins of domain-specific feature extraction and distance measure
- Easy construction for different data types

Digital Photos

Speech Recordings



3D Shape Models



Microarray Gene Expression Data

Image Spam

Conclusions

- Building efficient similarity search systems for large-scale feature-rich data
- Sketch construction
  - Compact metadata representation
  - Order of magnitude space reduction
- Multi-probe LSH indexing
  - Efficient similarity search in high dimensions
  - Order of magnitude space reduction
- Ferret toolkit
  - Built search systems for 6 different data types