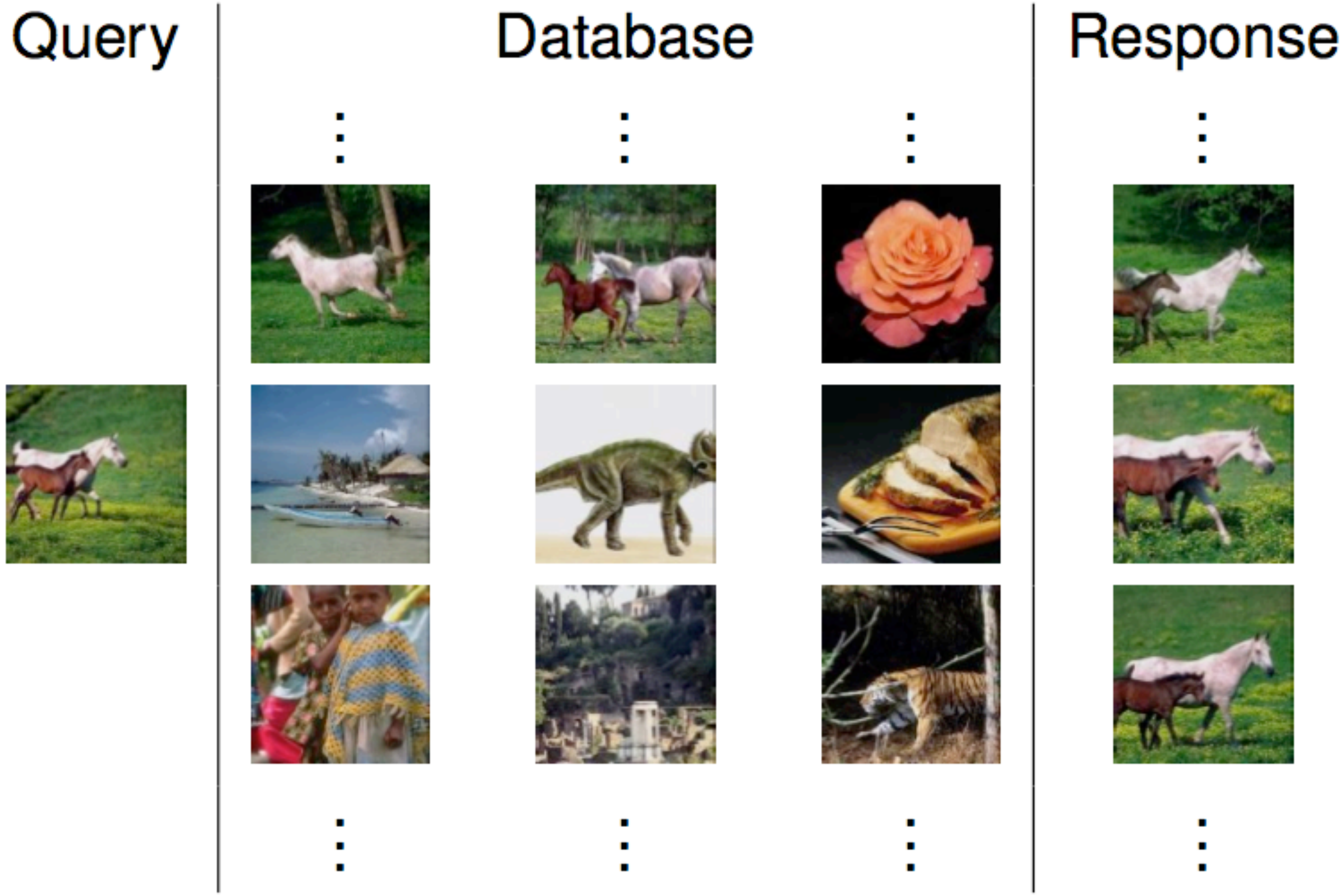
VQ Image Based Retrieval Using Color and Position Features

Ajay Daptardar and James A. Storer

Brandeis University

Content Based Image Retrieval (CBIR)



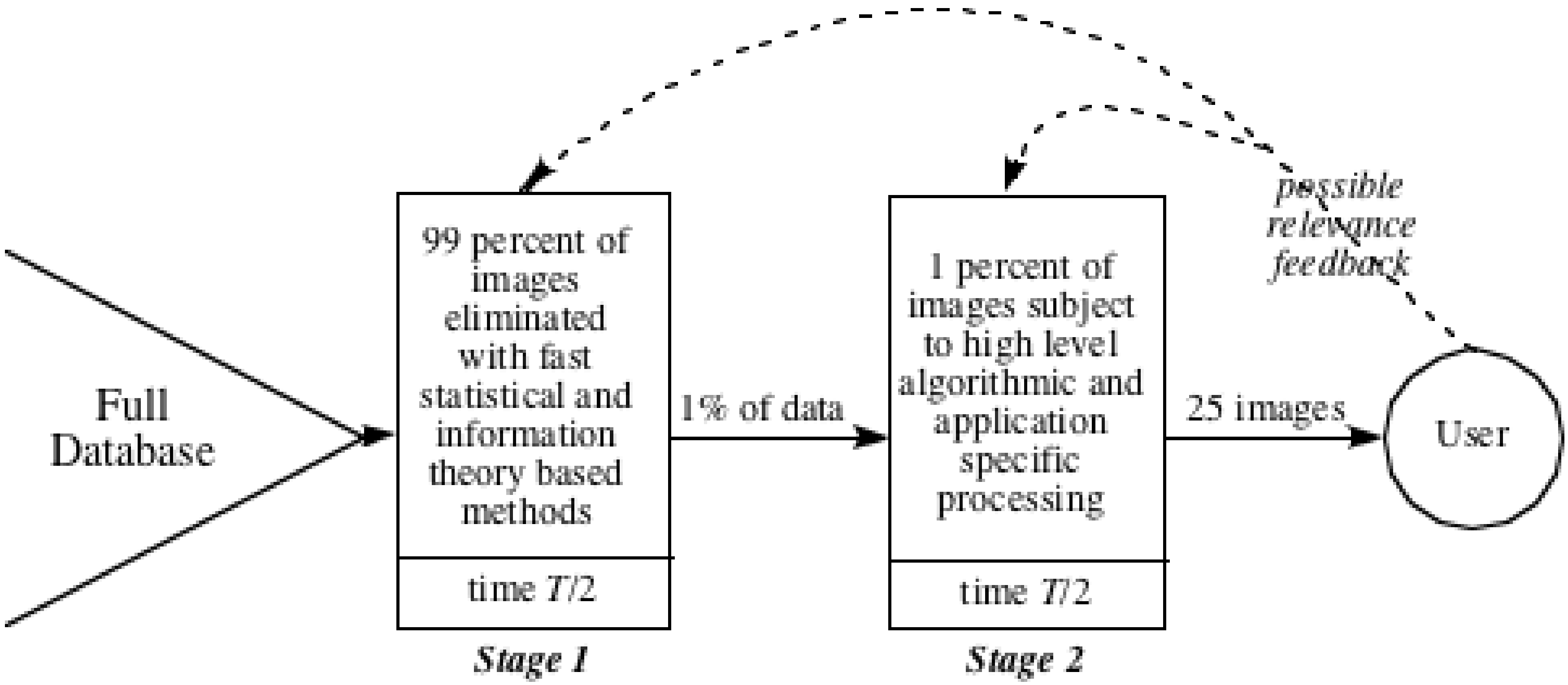
Basic "Query by Example" Problem:

Given an image, find others like it.

***** We do not address:***

- Additional user input (e.g., specification of regions of interest).
- Metadata associated with an image.
- Image context.
- Application specific processing.
- Relevance feedback.

Multi-Stage Processing



Previous Related Work

color histogram

(compute image similarity with color histogram intersection)

global VQ codebook

(compute image similarity with intersection between code vector usage histograms)

*** **minimum distortion image retrieval** (MDIR)

(encode query image with information based on the database image; less distortion means more similar)

This Work

- Motivated by the work of Jeong and Gray (DCC 2005), where Gaussian mixture models are employed for minimum distortion image retrieval (GMM-MDIR).
- We improve upon methods we presented in DCC 2006, that achieved comparable performance with lower complexity.

Simple VQ-Based Retrieval

Preprocess the database:

Sub-sample each image in the database to create a thumbnail, and construct a small VQ codebook for feature vectors derived from each thumbnail.

(i.e., each database image is tagged with a small codebook)

Given a query image:

Compress the query image with each codebook in the database and rank the images of the database in order of the achieved distortion.

Forward Compression Model

Database codebooks are used to compress the query image.

- Tends to perform better than ***backward compression***, where each image in the database is compressed with the codebook of the query image.
- Used by others such as Jeong and Gray (DCC 2005) to which we compare our work.
- Combining forward and backward compression in some way does not seem to provide enough benefit to justify the increased computation.

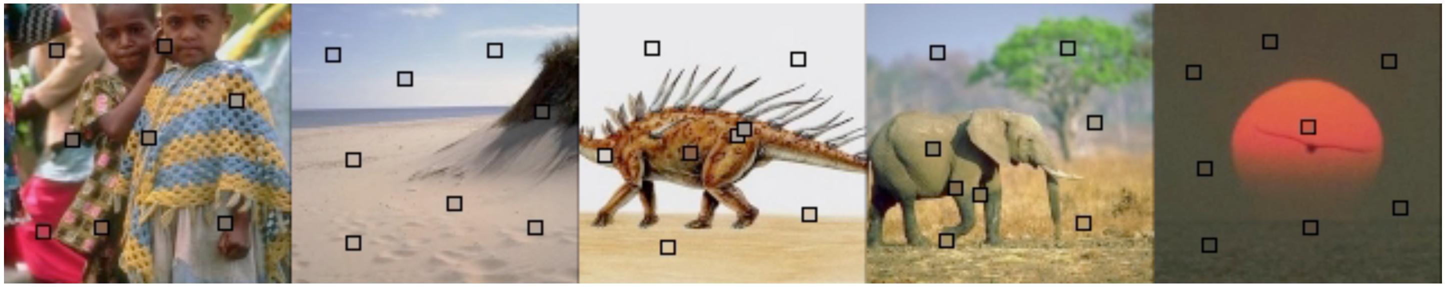
Color Feature Vectors

- Thumbnails are 128 by 128, using CIE-LUV color space.
- Feature vectors are formed from 2x2 image blocks.
- A (6 component) color feature vector consists of the mean and variance of each color channel:

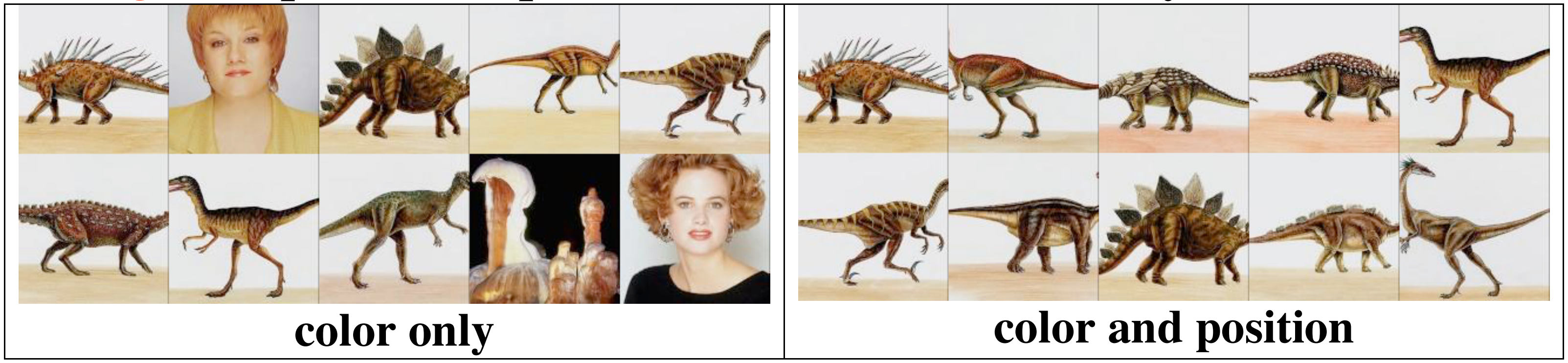
$$\begin{aligned} \mu_c &= \tfrac{1}{4} \sum_{i=0}^{1} \sum_{j=0}^{1} p_{ij}^{(c)}, \\ \sigma 2_c &= \tfrac{1}{4} \sum_{i=0}^{1} \sum_{j=0}^{1} \left(p_{ij}^{(c)} - \mu_c \right)^2 \end{aligned}$$

Our Previous Work

Each codebook entry consists of 8 components formed from 6 color feature components and the mean XY coordinates of all blocks associated with that entry.



Advantage: Captures simple structure that is fooled by color.



color only **color and position**

Disadvantage: Application specific ad-hoc weighting is used and feature vector size increased from 6 to 8.

Separate Training with Color-Position Codebooks

Training

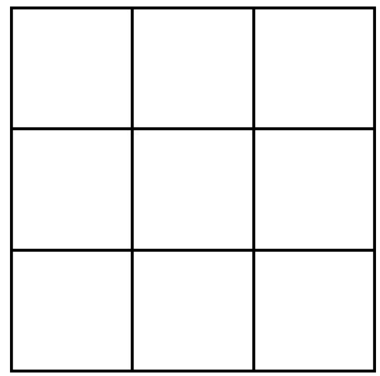
- Train a **color** VQ codebook using only the color components.
- Map training vectors to this codebook using only their color components.
- Using these as the new training set, train a **position** VQ codebook for this color.

Testing

- For each test vector, find the best color from the color codebook and then for that color find the best position.
- The total distortion is the sum of the color and position distortions.

Region Based Approach

Partition all images into regions, where the score for a pair of images is computed by summing the scores for each region. For all experiments reported here, 9 equal size regions are used:



*** Database TSVQ codebooks for each region are pruned to match the best sizes for the query image.

(Based on a single parameter, the ***query threshold***, codebooks for each query image region are trained so that MSE is less than this threshold.)

Region Based Retrieval Summary

0. Each image region has an associated 32 entry TSVQ codebook.

1. Choose a query threshold q.

2. **for** each region R of the query image Q **do**
   Construct a codebook that gives a MSE of less than q on R.

3. **for** each image I in the database **do begin**

    for each region R of I **do begin**

        Prune the codebook for R to be the same size as the corresponding codebook of the query image.

        (By repeatedly merging siblings u and v with parent w such that $D(w) - (D(u) + D(v))$ is minimal amongst all siblings, where $D(n)$ is the training distortion at node n, and is stored in the codebook.)

        Use R's codebook to compress the corresponding region of Q.

        end

    Sum MSEs for each region of Q to get a total score for I.

    end

4. Rank database images in order of increasing score.

Example Region Codebook Sizes

2	2	1
19	15	4
4	4	3

3	9	7
12	18	10
10	5	6

The COREL Database Used For Experiments

- 1500 JPEG images, 256x384 pixels each, organized into 15 classes of 100 images each (Wang Wiederhold [2001]).
- Used by Jeong and Gray (DCC 2005).

Precision V. Recall

a = number of relevant images retrieved (i.e., in the same class)

b = number of irrelevant images retrieved

c = number of relevant images NOT retrieved

Definition

Precision

$$P = \frac{a}{a+b}$$

as the fraction of the retrieved images that are relevant.

Definition

Recall

$$R = \frac{a}{a+c}$$

as the fraction of the relevant images that are retrieved.

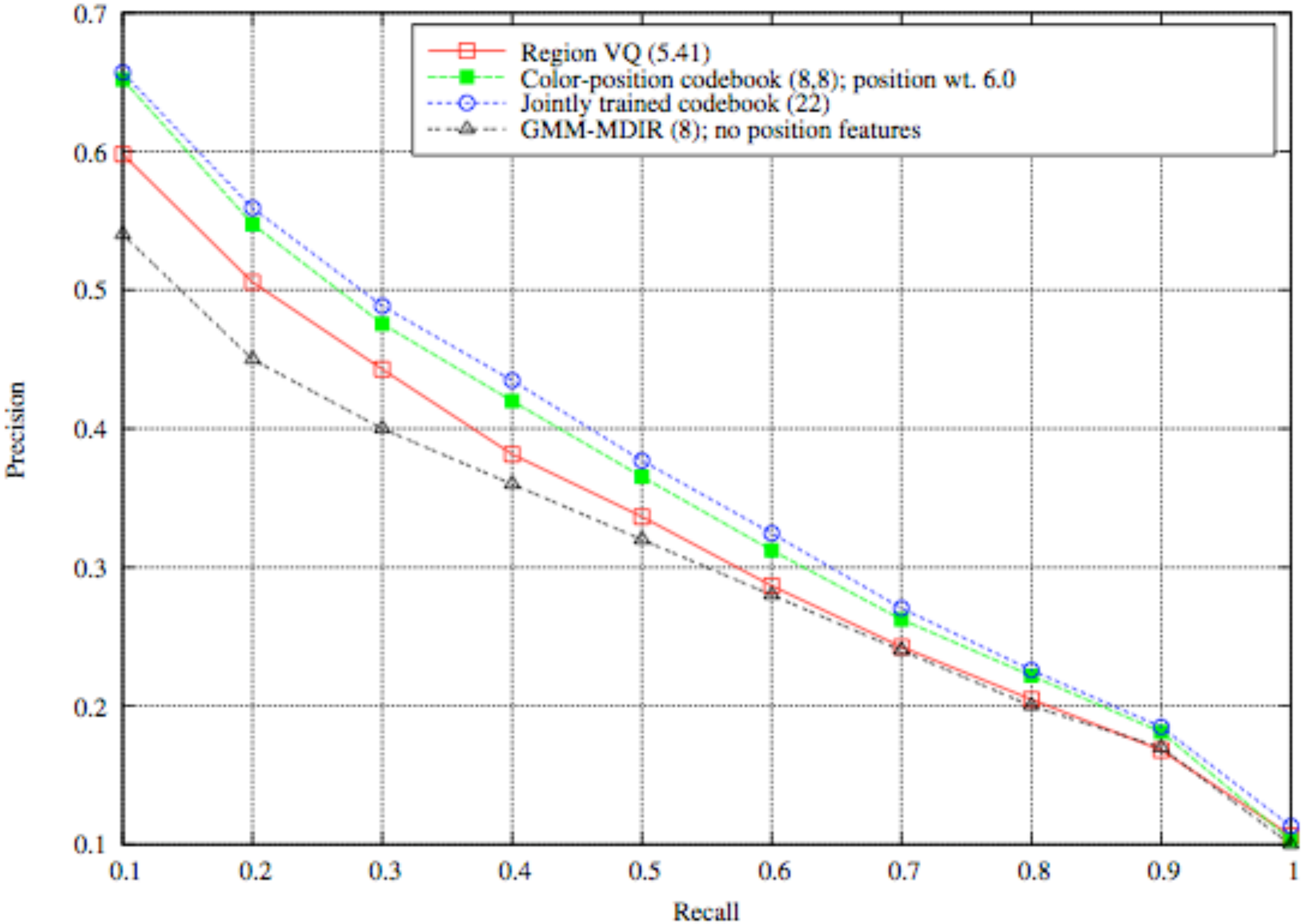
Experiments

We compare:

- Region VQ
- Color and position trained separately.
- Color and position trained jointly (as in our DCC 2006 paper).
- GMM-MDIR (Jeong and Gray DCC 2005).

* For compatibility with results presented by Jeong and Gray (DCC 2005) we present precision-recall plots for the same subset of 210 queries that they use (results for all 1500 queries are essentially the same).

Region VQ Compared to Previous Methods



Complexity

Consider the number of multiplications for an elementary step that searches for the best codeword in a codebook (i.e., this elementary step is repeated once for each block of the query image).

GMM-MDIR with 8 components per Gaussian mixture and full covariances uses one vector-matrix multiplication (8×8) and one inner product (8) for each of the 8 Gaussians for a total of:

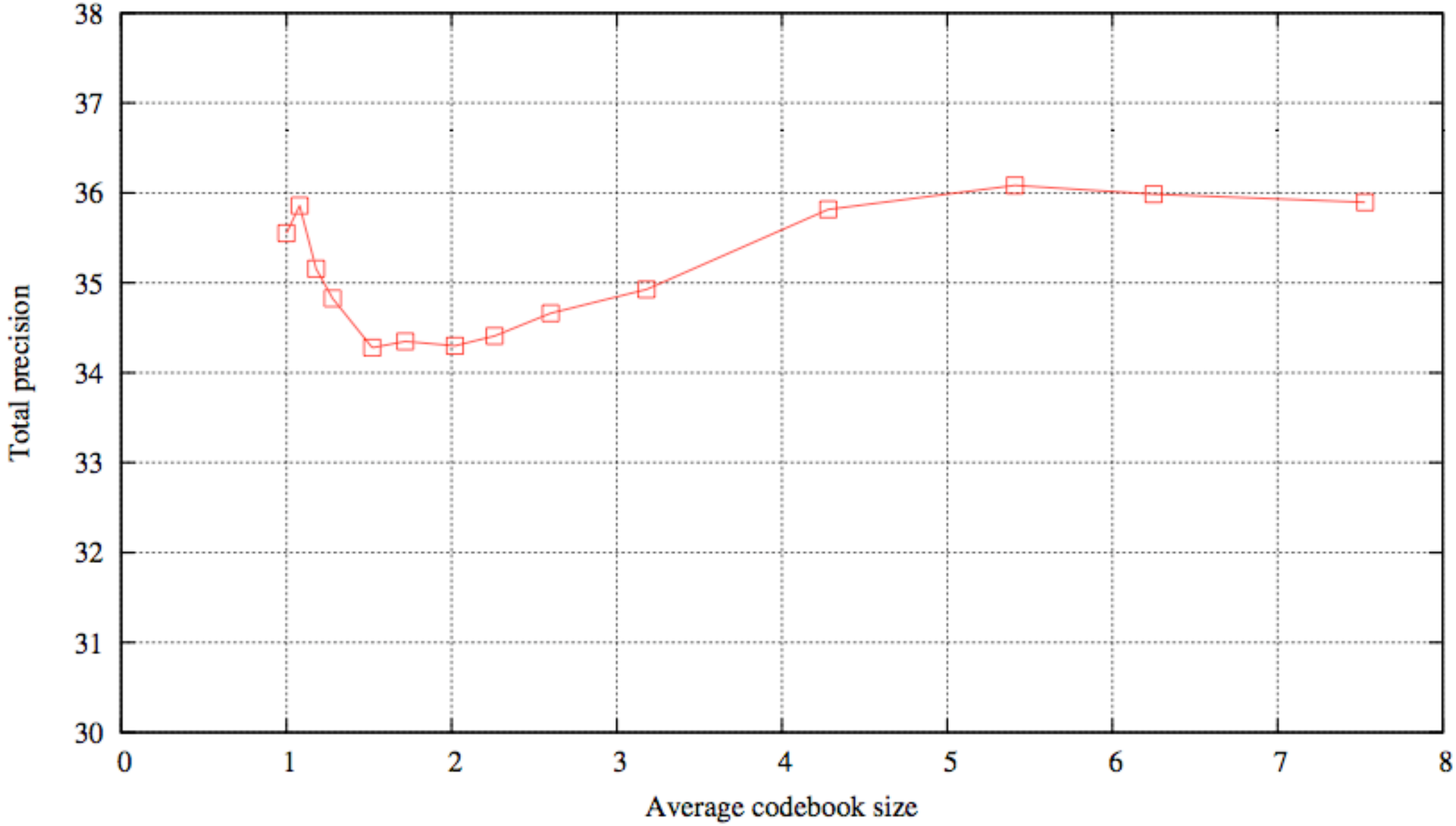
$8(8 \times 8 + 8) =$ **576 multiplications**

Region VQ with average codebook size 5.41 uses:

$5.41 \times 6 =$ **32.46 multiplications**

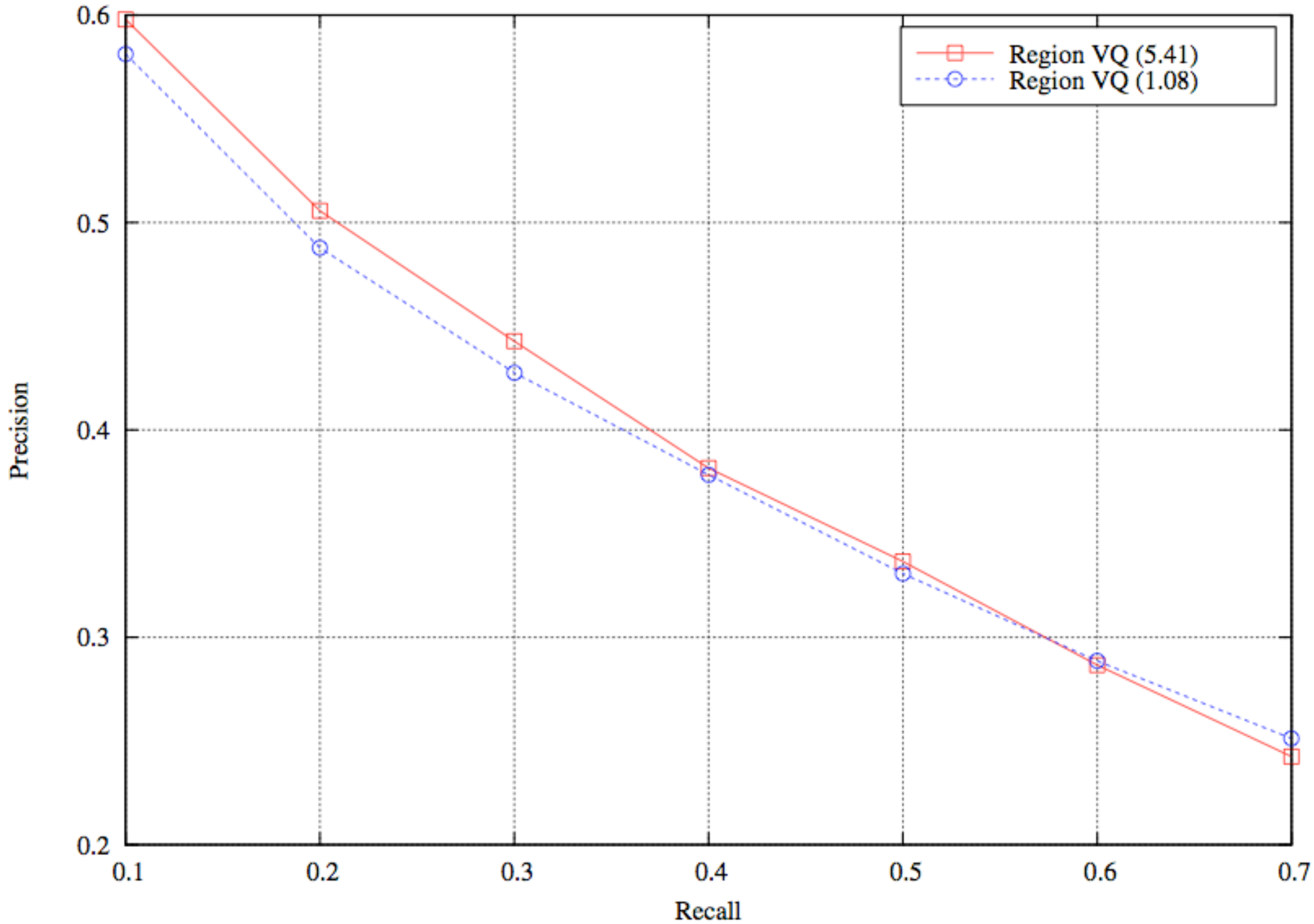
* Also, the computation for each image lends itself well to a simple parallel EREW model.

Precision V. Codebook Size



* Best precision is with average codebook size 5.41.

Reduced Codebook Size



* Average codebook size 1.08 is very close to best, and reduces an elementary step to $1.08 \times 6 =$ **6.48 multiplications**

Samples of "Good" Queries

Results

Results

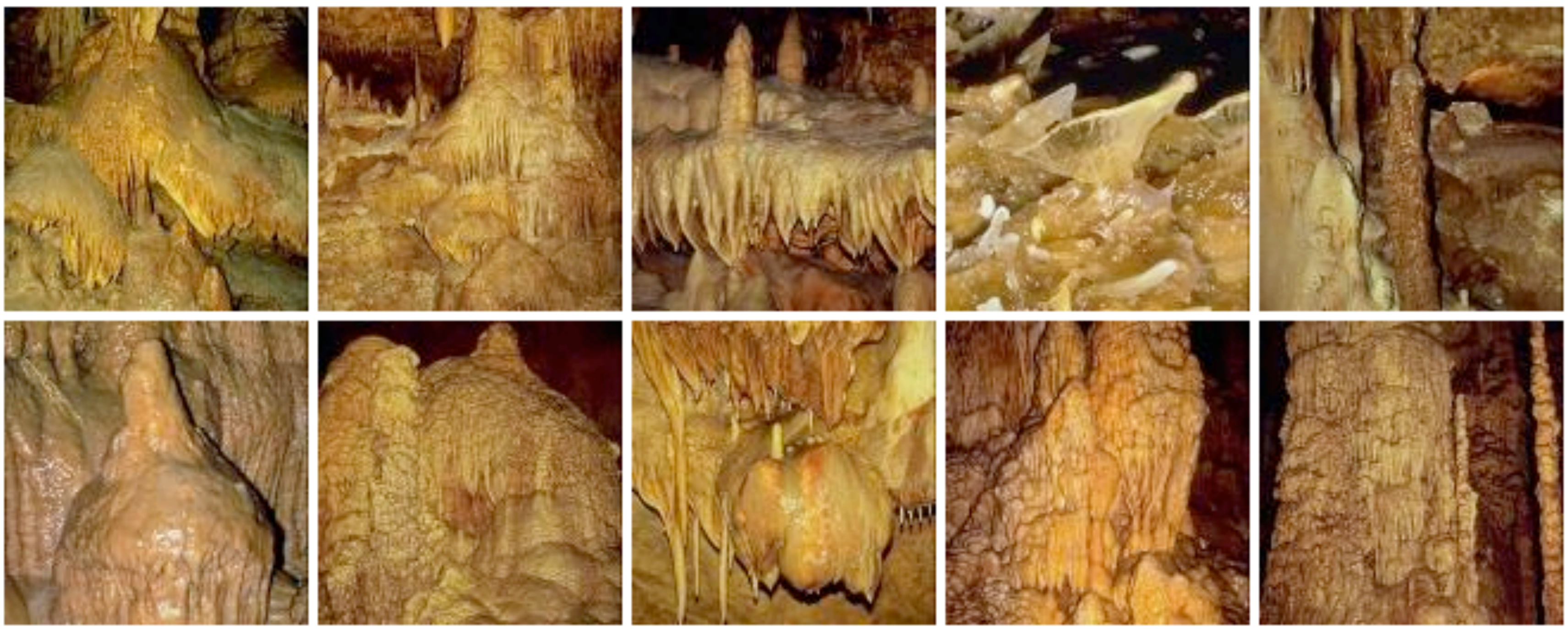
Results

Results

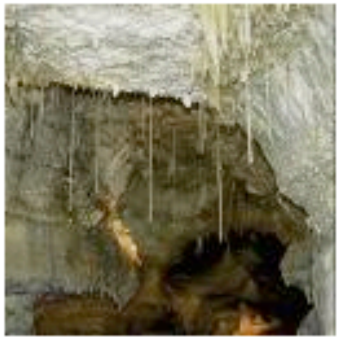
Results

Results

Results



Samples of "Bad" Queries



Results

Results

Reordering Based on Texture Analysis ?

Results

Other Areas of Future Research

- Adaptive region sizes (possibly with overlapping).
- Region adjustment based on relevance feedback.
- Retrieval based on regions of interest.
- Multiple query images.
- Other databases and better measures than precision-recall.

Thank you.