The
University
of Auckland

CBIR: Interaction & Evaluation

COMPSCI.708.S1.C

A/P Georgy Gimel'farb

The University of Auckland

Semantic vs. Feature Similarity

- The user seeks **semantic similarity**, but CBIR provides **similarity by data processing results**
- The challenge for a CBIR is to focus on a narrow information domain the user has in mind via specification, examples, and interaction
  - Early CBIR engines required from users to manually select low-level visual features and specify relative weights for each their possible representation

The University of Auckland

Early CBIR Engines

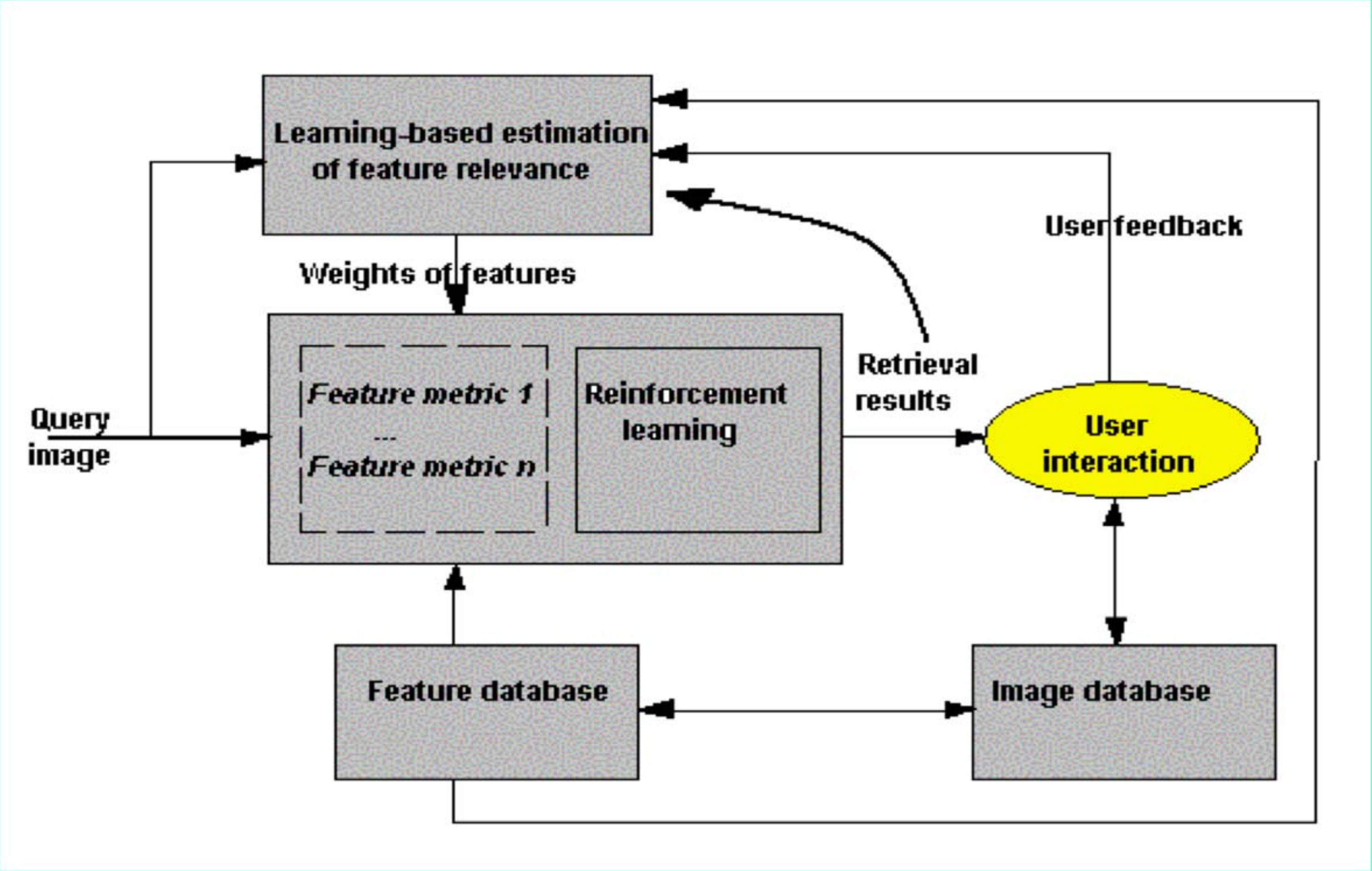
- Users had to know how the features are used
- Difficulties of representing semantic contents in terms of low-level features
  - Users need **semantics** (*"a sunset image", "penguins on icebergs*"), **rather than** general **low-level features** (*"a predominantly red/orange image", "predominantly oval black blobs on a white background*")
  - There exist too many irrelevant images with similar dominant colours and regions (a "retrieval noise")
  - Difficulties by the highly subjective human perception

The University of Auckland

More Advanced CBIR Engines

- Low-level features are not adequate to contents
- ***Subjective perception***: different users and even the same user under different conditions may interpret the same image differently
- Visually similar images: due to their semantics, rather than their similar low-level features
  - Experimental CBIR engines (e.g. Photobook with FourEyes or PicHunter) use **relevance feedback** to adjust a query in such a way as to approach close to the user's expectations

Interactive CBIR Engine



The University of Auckland

Interactive CBIR Engine

- An interactive CBIR system contains:
  - an image database
  - a feature database
  - a selector of feature similarity metric
  - a block for evaluating feature relevance
- When a query arrives, the system has no prior knowledge about the query: all features have the same weight in computing the similarity measure

The University of Auckland

Interactive CBIR Engine

- After a fixed number of the top-rank (by the similarity to the query) images are retrieved, the user provides the *relevance feedback*
- The **feature relevance block** uses learning algorithms in order to re-evaluate the weights of each feature in line with the user's feedback
- The **metric selector** chooses the best similarity metric for the weighted features using reinforcement learning

The University of Auckland

Interactive CBIR Engine

- By iteratively using the relevance feedback, the engine adjusts the query and brings the retrieved images closer to the user's expectations
  - The weight of each feature in the similarity computation is iteratively updated in accord with the high-level and subjective human perception
- The user need not map semantics onto features and specify weights and instead only informs the engine which images are relevant to the query

The University of Auckland

Interactive QBE Retrieval

- Two-stage process of formulating a query:
  - an **initial formulation** when the user has no precise idea of what should be searched for
  - a **refined formulation** after the user took part in the iterative process of the relevance feedback
- First stage: the engine helps in formulating an "imprecise" query by providing sequential and feature-based browsing and sketching tools

Interactive QBE Retrieval

- Second stage: the user gives positive and negative feedback to the system
- Feedback: (1) all currently retrieved images are labelled in accord with their relevance to user's expectations
  - E.g. image labelling into five groups: *highly relevant, relevant, neutral, irrelevant*, and *highly irrelevant* results of the retrieval

The University of Auckland

Interactive QBE Retrieval

- Feedback: (2) The CBIR system processes both the query and the user-labelled retrieved images
  - The joint processing updates weights of features and chooses more adequate similarity metric
  - **The goal of processing**: to suppress the irrelevant outputs and enhance the relevant ones
    - If the range of feature values for the relevant images is similar to that for the irrelevant ones, then this feature cannot effectively separate these images and its weight should decrease
    - But if the "relevant" values vary in a relatively small range containing no or almost no "irrelevant" values, it is a crucial feature which weight should increase

The University of Auckland

How To Evaluate Retrieval?

Items	*Relevant*	*Non-relevant*
Retrieved	***A*** : hits	***B***: Noise, or fallout
Not retrieved	***C***: misses	***D***: Correct rejection

Effectiveness of retrieval depend on the **filtering capacity** of the system, i.e. on proportions of relevant and non-relevant items among the retrieved data and with respect to the whole data base

Evaluation of the QBE Retrieval

- Test-bed for the evaluation:
  - a collection of N images
  - a set of benchmark queries to the test bed data
  - the "ground-truth" quantitative assessment of the relevance of each image for each benchmark query
- Retrieval performance:
  - average *recall* / *precision*, i.e. *a*verage relative numbers of the relevant results returned to the user in all the benchmark queries

Evaluation of the QBE Retrieval

- Let $W_r \in [0,1]$ be a quantitative relevance of the item of rank r to the benchmark query
- For each cut-off value $n \in [1,N]$ of returns:
  - $A_n = W_1 + \ldots + W_n$ $\rightarrow$ returned relevant results
  - $B_n = n - A_n$ $\rightarrow$ returned irrelevant results
  - $C_n = W_{n+1} + \ldots + W_N$ $\rightarrow$ non-returned relevant results
  - $D_n = N - n - C_n$ $\rightarrow$ non-returned irrelevant results

Evaluation of the QBE Retrieval

- Recall $R_n = A_n / (A_n + C_n)$ is a relative amount of the relevant results returned among the n top-rank matches after a query
  - Recall by itself is not a good quality measure (as $R_N = 1.0$)
  - *Example*: N=10 database images; $n = 3$ images returned;

    W_1=0.9; W_2=0.8; W_3=0.7; $W_4 \ldots W_6$=0.4, $W_7 \ldots W_{10}$=0.2 – the relevance of the images ranked w.r.t. a query:

    $A_3 = W_1 + W_2 + W_3 = 0.9 + 0.8 + 0.7 = 2.4$

    $C_3 = W_4 + \ldots + W_{10} = 0.4 \times 3 + 0.2 \times 4 = 2.0 \rightarrow$

    $\boldsymbol{R_3} = 2.4 / (2.4 + 2.0) = 2.4 / 4.4 = \mathbf{0.545}$

Evaluation of the QBE Retrieval

The University of Auckland

- **Precision** $P_n = A_n / n$ is a proportion of relevant results returned among the n top-rank matches after a query
  - Precision is the average relevance of the returned results
  - *Example*: N=10 database images; $n = 3$ images returned; W_1=0.9; W_2=0.8; W_3=0.7; $W_4 \dots W_6$=0.4, $W_7 \dots W_{10}$=0.2 – the relevance of the images ranked w.r.t. a query:
    $A_3 = W_1 + W_2 + W_3 = 0.9 + 0.8 + 0.7 = 2.4 \rightarrow$
    $\boldsymbol{P_3} = 2.4 / 3 = \mathbf{0.8}$
  - **Precision–recall graph** depicts the degradation of precision at n as one traverses the output list

Evaluation of the QBE Retrieval

- Fallout $F_n = B_n/(B_n+D_n) = (n-A_n)/(N-A_n-C_n)$ is the relative amount of retrieved irrelevant items
  - It measures how quickly precision drops as recall increases
  - *Example*: N=10 database images; n = 3 images returned;

    W_1=0.9; W_2=0.8; W_3=0.7; $W_4 \ldots W_6$=0.4, $W_7 \ldots W_{10}$=0.2 – the relevance of the images ranked w.r.t. a query:

$$A_3 = W_1 + W_2 + W_3 = 0.9 + 0.8 + 0.7 = 2.4$$

$$C_3 = W_4 + \ldots + W_{10} = 0.4 \times 3 + 0.2 \times 4 = 2.0 \rightarrow$$

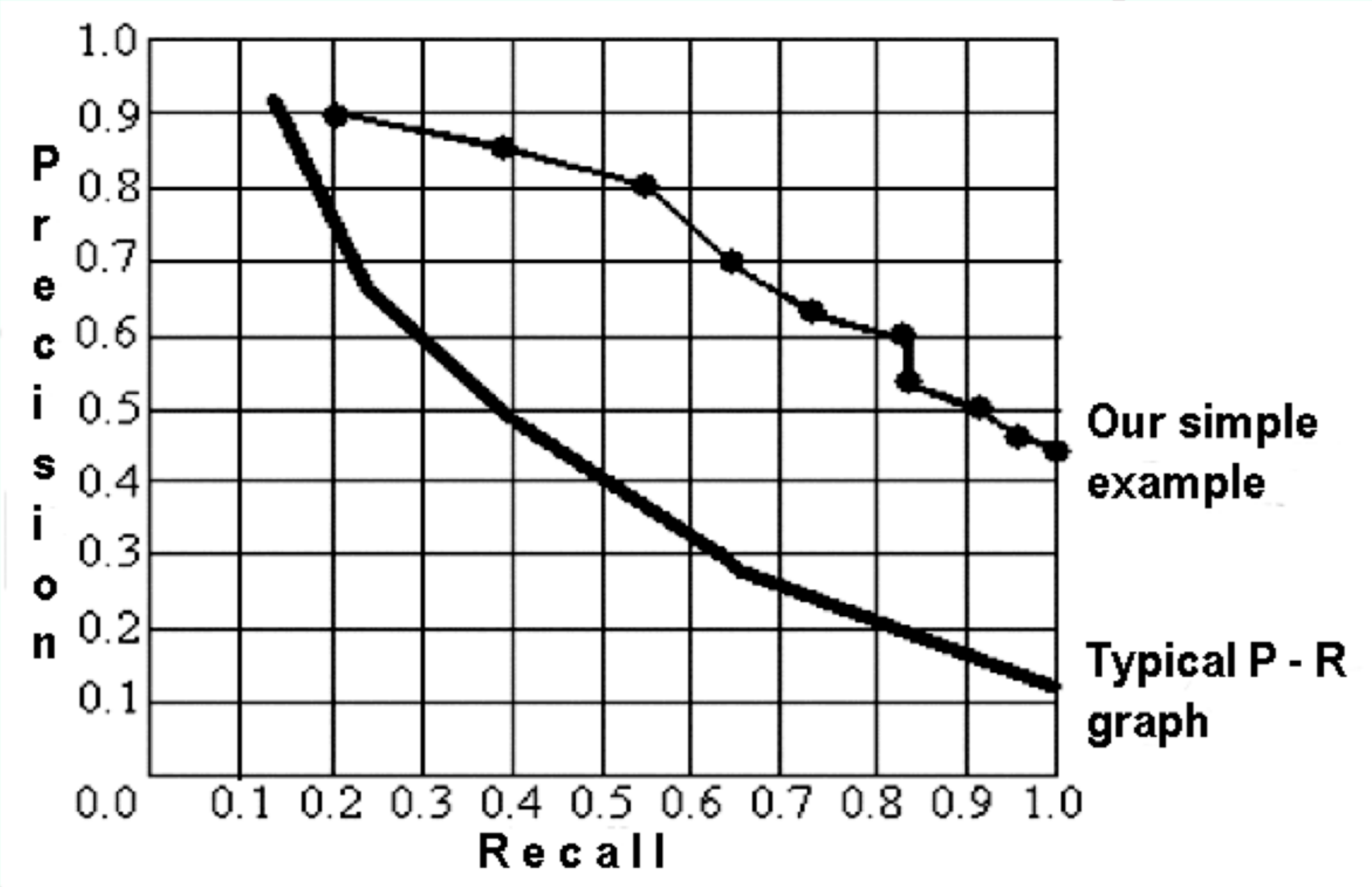
$$B_3 = 3 - 2.4 = 0.6; D_3 = 10 - 3 - 2.0 = 5.0 \rightarrow$$

$$\boldsymbol{F_3} = 0.6 / (0.6 + 5.0) = 0.6 / 5.6 = \mathbf{0.107}$$

Evaluation for n Top-rank Items

n	1	2	3	4	5	6	7	8	9	10
W_n	0.90	0.80	0.70	0.40	0.40	0.40	0.20	0.20	0.20	0.20
A_n	0.90	1.70	2.40	2.80	3.20	3.60	3.80	4.00	4.20	4.40
C_n	3.50	2.70	2.00	1.60	1.20	0.80	0.60	0.40	0.20	0.00
$\boldsymbol{R_n}$	**0.20**	**0.39**	**0.55**	**0.64**	**0.73**	**0.82**	**0.83**	**0.91**	**0.96**	**1.00**
$\boldsymbol{P_n}$	**0.90**	**0.85**	**0.80**	**0.70**	**0.64**	**0.60**	**0.54**	**0.50**	**0.47**	**0.44**
B_n	0.10	0.30	0.60	1.20	1.80	2.40	3.20	4.00	4.80	5.60
D_n	5.50	5.30	5.00	4.40	3.80	4.20	2.40	1.60	0.80	0.00
$\boldsymbol{F_n}$	**0.02**	**0.05**	**0.11**	**0.21**	**0.32**	**0.43**	**0.57**	**0.71**	**0.86**	**1.00**

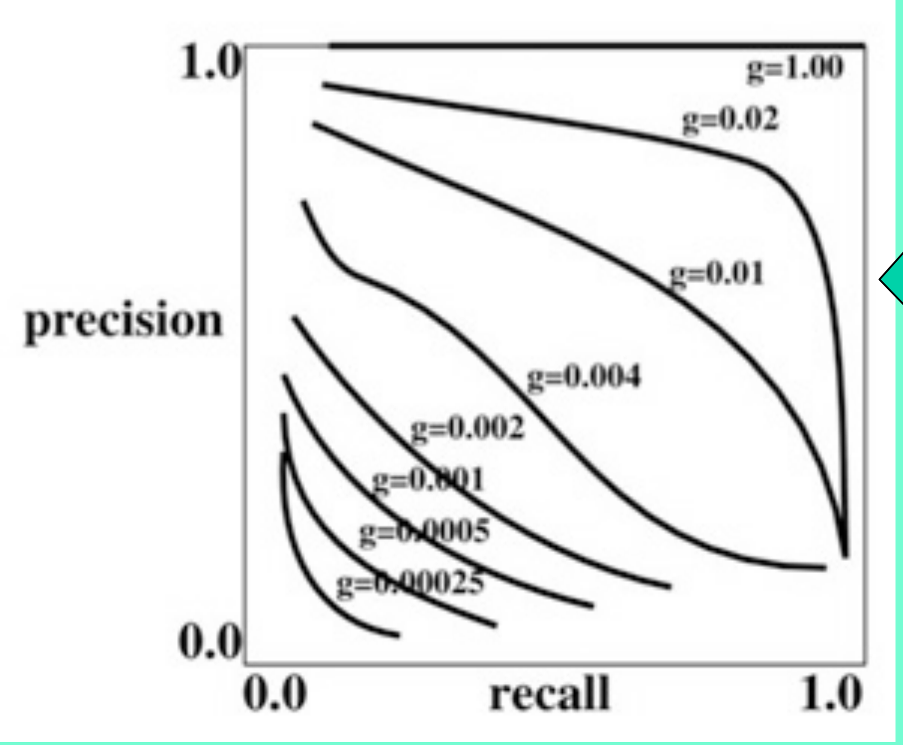
Precision – Recall Graph



Generality Vs. Performance

- Precision - Recall graphs are meaningful only if their points are measured under a common generality: $G=(A_n + C_n)/N$

G is the common average expected performance



Typical P-R curves for retrieving a constant-size group of totally relevant items embedded in a growing number of irrelevant items

In practice, no complete ground truth to evaluate recall and generality is known; only their lower bounds $A_n/(N-n-A_n)$ and A_n/N can be used to analyse a CBIR system