

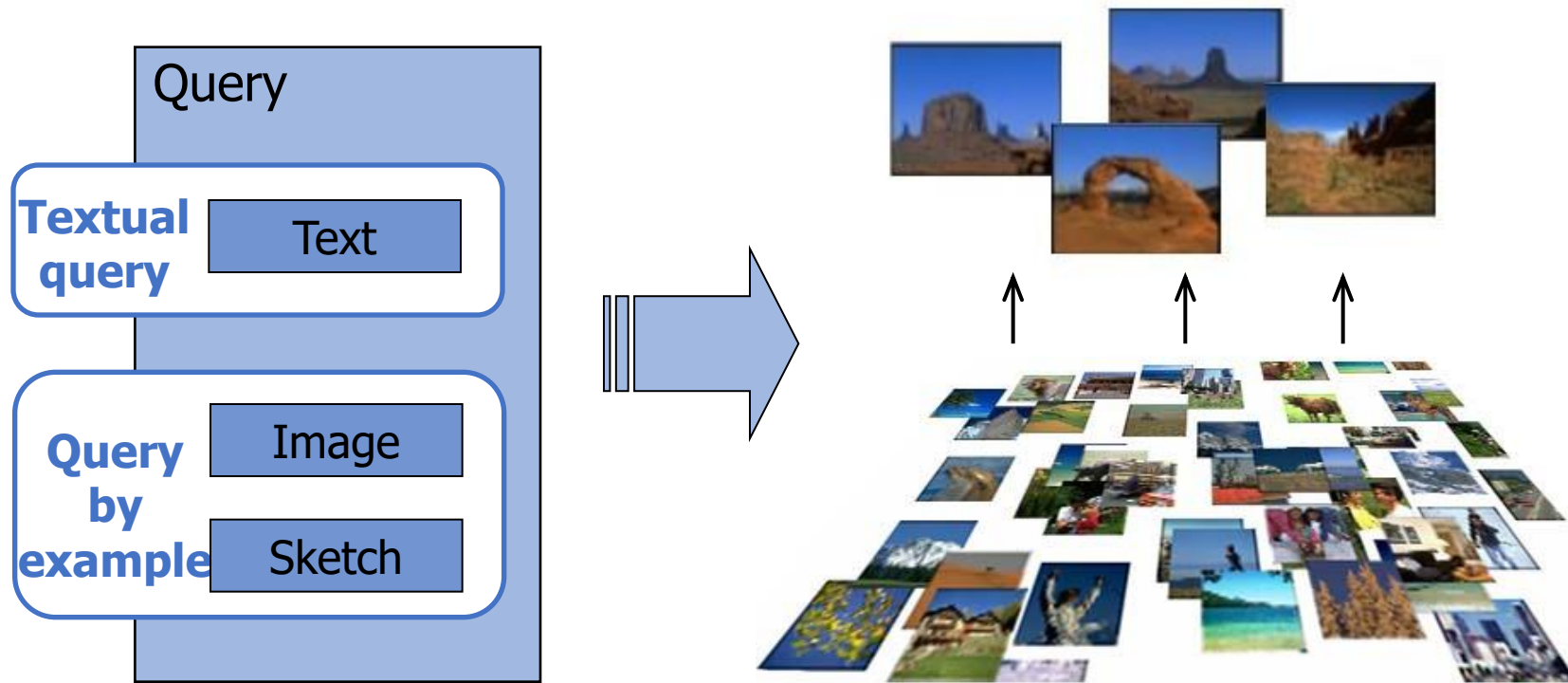
# Content-based Image Retrieval

# Lecture Outline

- What is and Why image retrieval?
- How to compare and retrieve images?
  - Common components of the CBIR systems
  - Main problems and research directions
- Performance measurement
  - Retrieval effectiveness

# What is image retrieval?

- Description Based Image Retrieval (DBIR)
- Content Based Image Retrieval (CBIR)



# DBIR vs. CBIR

	DBIR	CBIR
+	<ul style="list-style-type: none"><li>▪ Fulltext search algorithms are applicable</li><li>▪ Search results corresponds to image semantics</li></ul>	<ul style="list-style-type: none"><li>▪ Automatic index construction</li><li>▪ Index is objective</li></ul>
-	<ul style="list-style-type: none"><li>▪ Manual annotating is hardly feasible</li><li>▪ Manual annotations are subjective</li></ul>	<ul style="list-style-type: none"><li>▪ Semantic gap</li><li>▪ Querying by example is not convenient for a user</li></ul>

# Levels of image retrieval

- Level 1: Based on color, texture, shape features
  - Images are compared based on low-level features, no semantics involved
  - A lot of research done, is a feasible task
- Level 2: Bring semantic meanings into the search
  - e.g., identifying human beings, horses, trees, beaches
  - Requires retrieval techniques of level 1
  - Very active and challengeable research area
- Level 3: Retrieval with abstract and subjective attributes
  - Find pictures of a particular birthday celebration
  - Find a picture of a happy beautiful woman
  - Requires retrieval techniques of level 2 and very complex logic

# Why image retrieval?

- Huge amounts of images are everywhere: how to manage this data?
- “A Picture is worth thousand words”
- Not everything can be described in text

# Why content-based image retrieval?

- Automatic generation of textual annotations for a wide spectrum of images is not feasible.
- Annotating images manually is a cumbersome and expensive task for large image databases.
- Manual annotations are often subjective, context-sensitive and incomplete.
- Google, Yandex and others use text-based search. Results are not perfect.

However, now it is much better, than a couple of years ago!

# Image retrieval by Google

Google window

Поиск картинок Поиск в Интернете Расширенный поиск картинок

Картинки Показывать: Все размеры Результаты 1 - 21 из примерно 67 800 000 для window. (0,15 с)

музыка, I. Mad Heads - Evil People ...  
744 x 608 - 147k - jpg  
flaur-des12gn.livejournal.com

Window: Окно( бумага, тушь)  
826 x 1169 - 110k - jpg  
eadam99.googlepages.com

... /window/window-640x480.jpg  
640 x 480 - 121k - jpg  
www.artlebedev.ru

Aphex Twin - Window Licker  
516 x 342 - 25k - jpg  
klipakov.net

A Better Replacement Window  
580 x 333 - 132k - jpg  
www.andersenwindows.com

Change Window Title Bar Text/icon  
1004 x 645 - 161k - jpg  
inetlive.ru

Kitchen window coverings  
704 x 322 - 31k - jpg  
www.budgetblinds.com

Recessed Window  
1200 x 1600 - 567k - jpg  
dailyhomenotips.com

By modifying the window size setting  
448 x 316 - 20k - jpg  
www.mydigitalife.info

PowerShell window  
572 x 330 - 10k - png  
blog-not-a-kemel-guy.com

Open Window - Open Window  
[2006]  
300 x 300 - 26k - jpg  
funky4souls.com

Free Window Sweeper так же имеет  
450 x 338 - 45k - jpg  
onhold.ru

Концептуальная схема X Window  
600 x 428 - 22k - gif  
www.i2r.ru

Figure 3-5 Window List GUI Ti  
357 x 292 - 22k - gif  
docs.sun.com  
[ Еще с домена docs.sun.co

Window Treatments  
468 x 468 - 36k - jpg  
www.trendir.com

Stardock Window Blinds - 310 ...  
500 x 382 - 64k  
www.cwer.ru

Power Window Regulator, power  
window ...  
450 x 450 - 65k - jpg  
www.taiwan.ru

My Windows 2007  
571 x 799 - 218k - jpg  
meskhi.livejournal.com

Report window  
676 x 859 - 24k - gif  
docs.sun.com

... Black Window LANBOX 4i  
352 x 352 - 15k  
images.digitalshop.ru

Googoooooooooooooogle

1 2 3 4 5 6 7 8 9 10 Следующая



# Image retrieval by Yandex

**Яндекс**  
Найдётся всё

**картинки**

Почта | Мои находки | Настройки | Войти

**window** расширенный поиск Найти

Всё | Блоги | Картинки | Все службы

Любые | Обыч | Большие | Средние | Маленькие | Черно-белые | Цветные

Найдено картинок: **883 385**, сайтов: **76 514**  
Включен умренный фильтр

**Яндекс.Фотки**

**Windows**  
[Еще по запросу «window»](#) 2187

**Window.**  
639×800, 202 KB, JPEG  
[www.artussia.ru](#) [Еще 48](#)

**Window**  
526×600, 39 KB, JPEG  
[www.photographic.com.ua](#) [Еще 38](#)

**Window**  
458×613, 51 KB, JPEG  
[darknsk.com](#) [Еще 2](#)

**window.jpg**  
500×486, 53 KB, JPEG  
[forum.poezha.net](#)

**Windows**  
533×800, 53 KB, JPEG  
[fotki.yandex.ru](#)  
[Еще 1532](#)

**Window "Across the Universe"**  
427×640, 100 KB, JPEG  
[www.ringofstars.ru](#) [Еще 2](#)

**window.jpg**  
650×650, 60 KB, JPEG  
[action.by](#)

**Andrei Marhotin - window**  
498×750, 119 KB, JPEG  
[www.marhotin.ru](#) [Еще 2](#)

**window**  
1120×840, 119 KB, JPEG  
[www.moldova.net](#) [Еще 2](#)

**Window**  
1024×768, 33 KB, JPEG  
[atlantic-photo.narod.ru](#)

**window**  
420×570, 98 KB, JPEG  
[www.eduard.ru](#) [Еще 2](#)

**Window**  
640×480, 110 KB, JPEG  
[russian.wunderground.co](#)  
[Еще 182](#)

**Window?**  
499×748, 107 KB

**Window**  
700×900, 114 KB, JPEG

**Window**  
1024×768, 127 KB, JPEG

**Window**  
433×620, 126 KB, JPEG

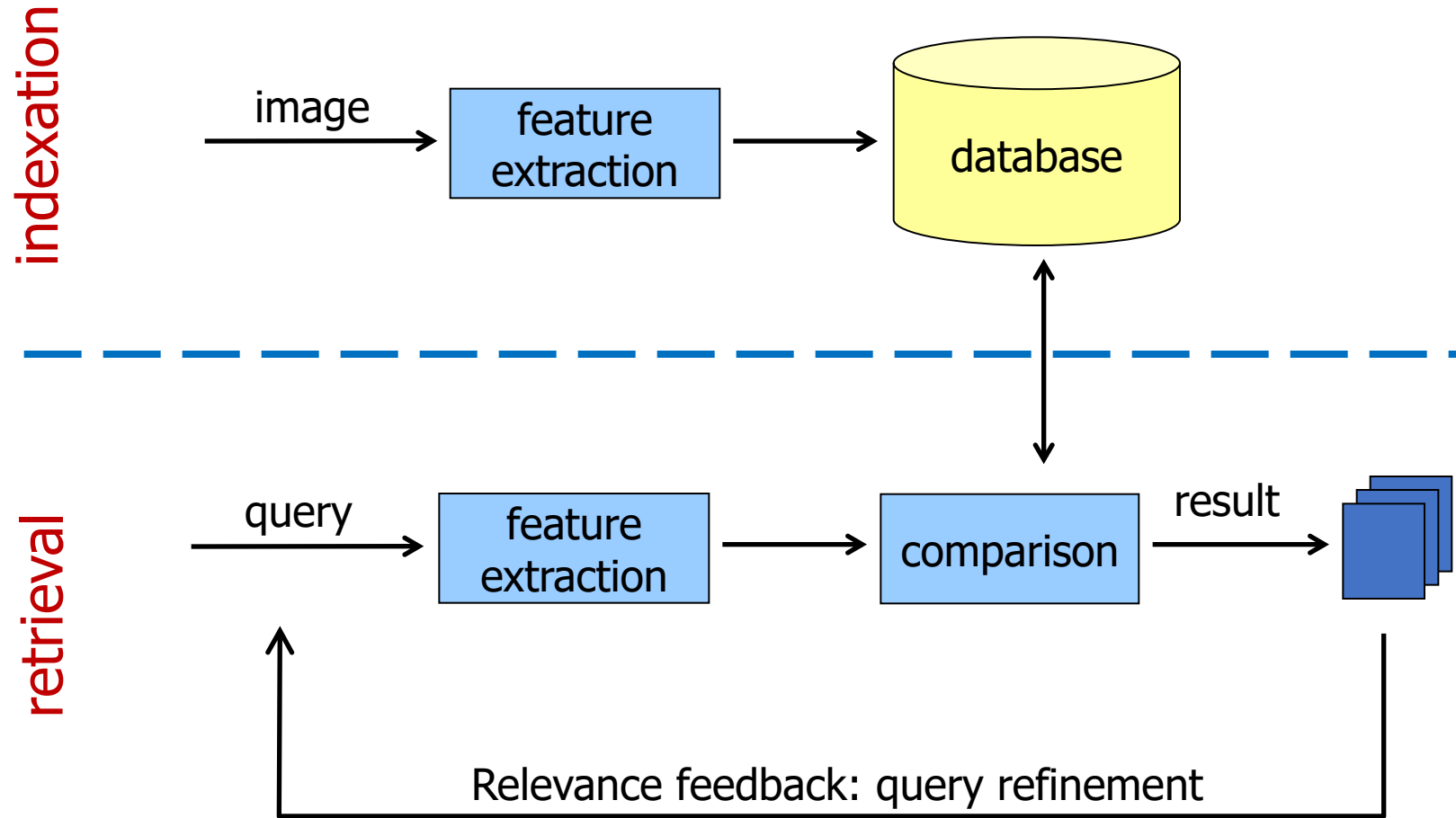
**Window**  
1024×627, 276 KB, JPEG

**window.jpg**  
1194×906, 35 KB, JPEG

# Lecture Outline

- What is and Why image retrieval?
- How to compare and retrieve images?
  - Common components of the CBIR systems
  - Main problems and research directions
- What are applications?
- Performance measurement
  - Retrieval effectiveness

# Common components of CBIR system



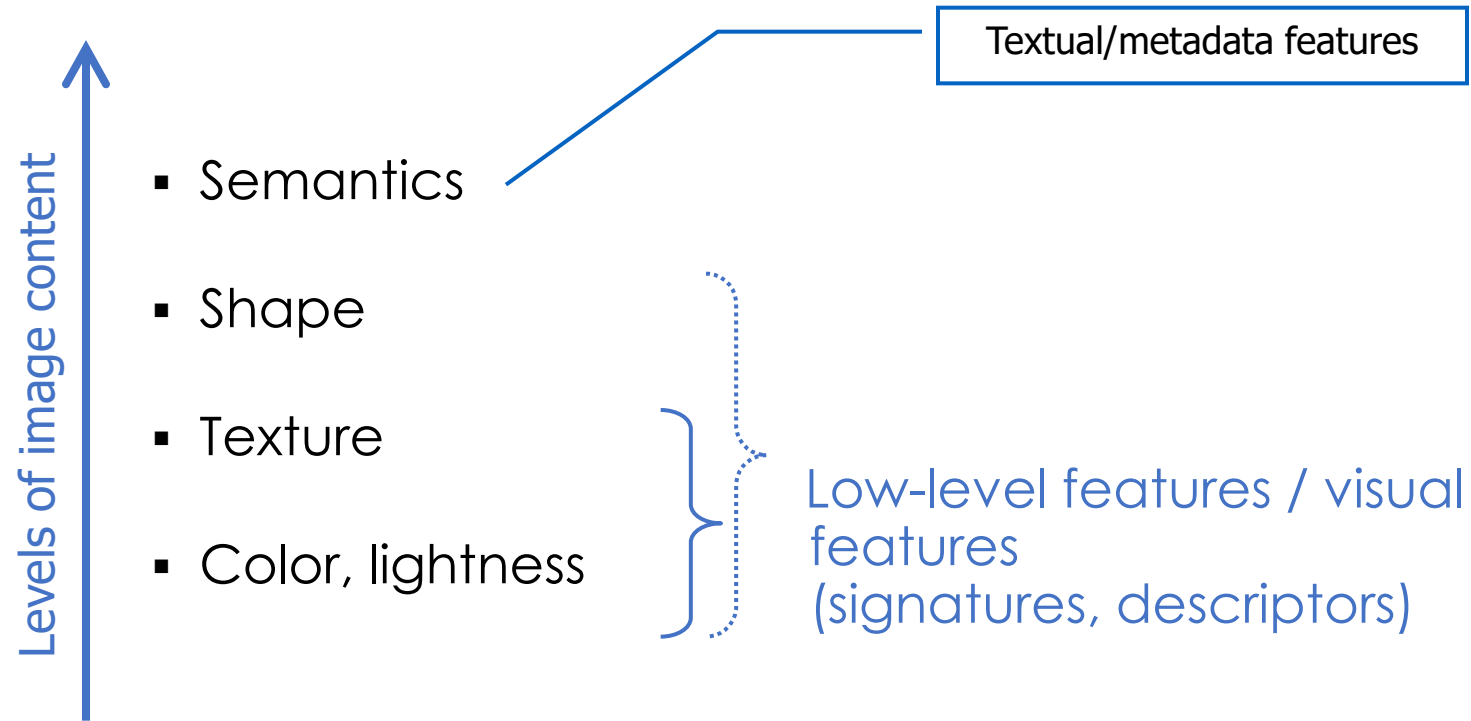
# Lecture Outline

- What is and Why image retrieval?
- How to compare and retrieve images?
  - Common components of the CBIR systems
  - Main problems and research directions
- Performance measurement
  - Retrieval effectiveness

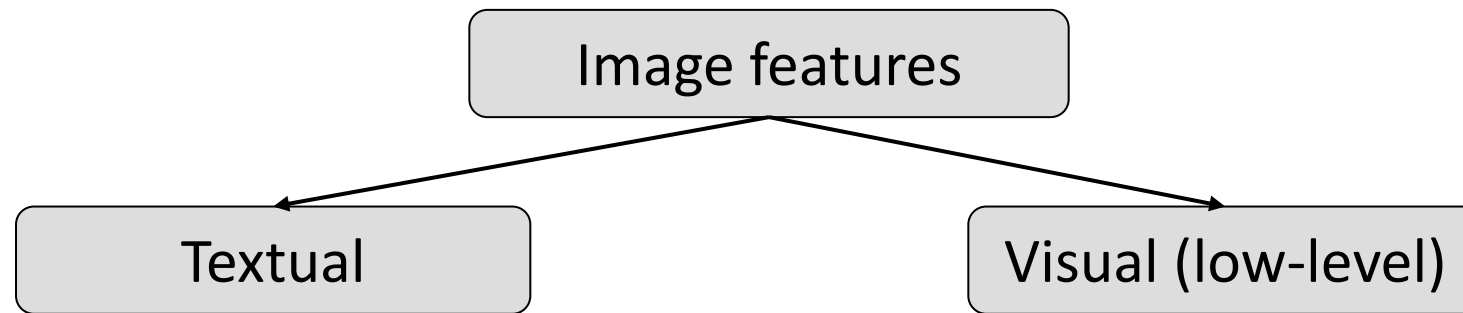
# Problems and directions

- Feature extraction
  - How to represent an image in a compact and descriptive way?
  - How to compare features, and, thus, images?

# How to: Image features



# How to: Image features



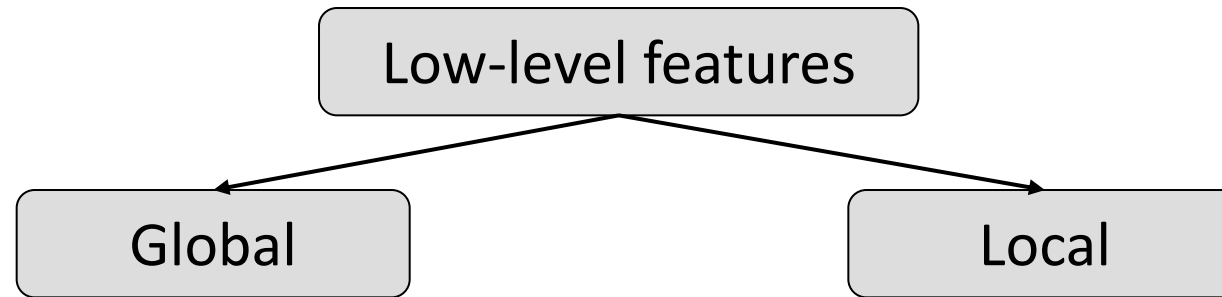
Annotations and metadata:

- tags/keywords;
- creation date;
- geo tags;
- name of the file;
- photography conditions (exposition, aperture, flash...).

Features extracted from pixel values:

- color descriptors;
- texture descriptors;
- shape descriptors;
- spatial layout descriptors.

# How to: Image features



Describes **the whole image**:

- average intensity;
- average amount of red;
- ...

All pixels of the image are processed.

Describes **one part** of the image:

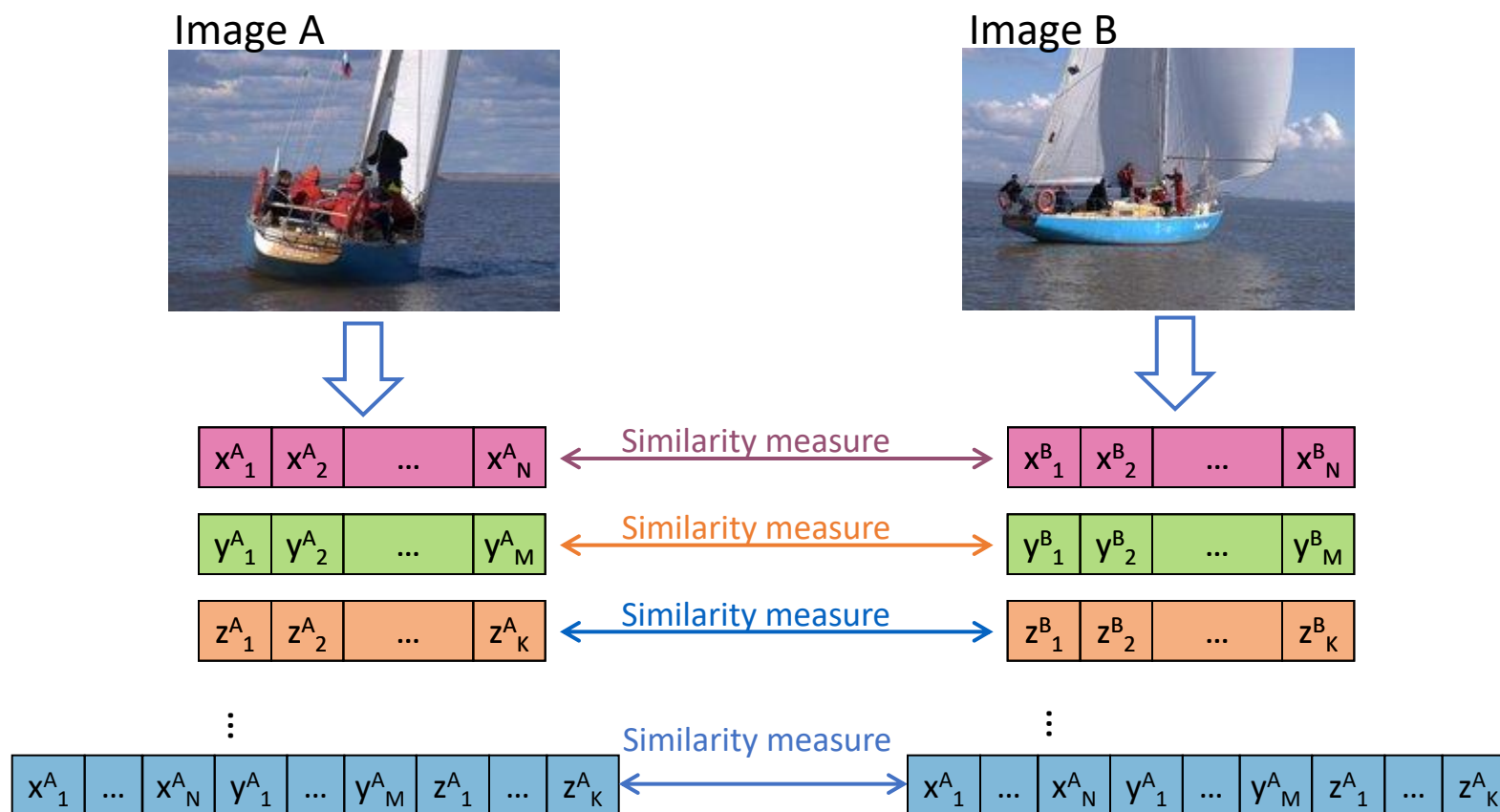
- average intensity for the left upper part;
- average amount of red in the center of the image;
- ...

Segmentation of the image is performed, pixels of a particular segment are processed to extract features.

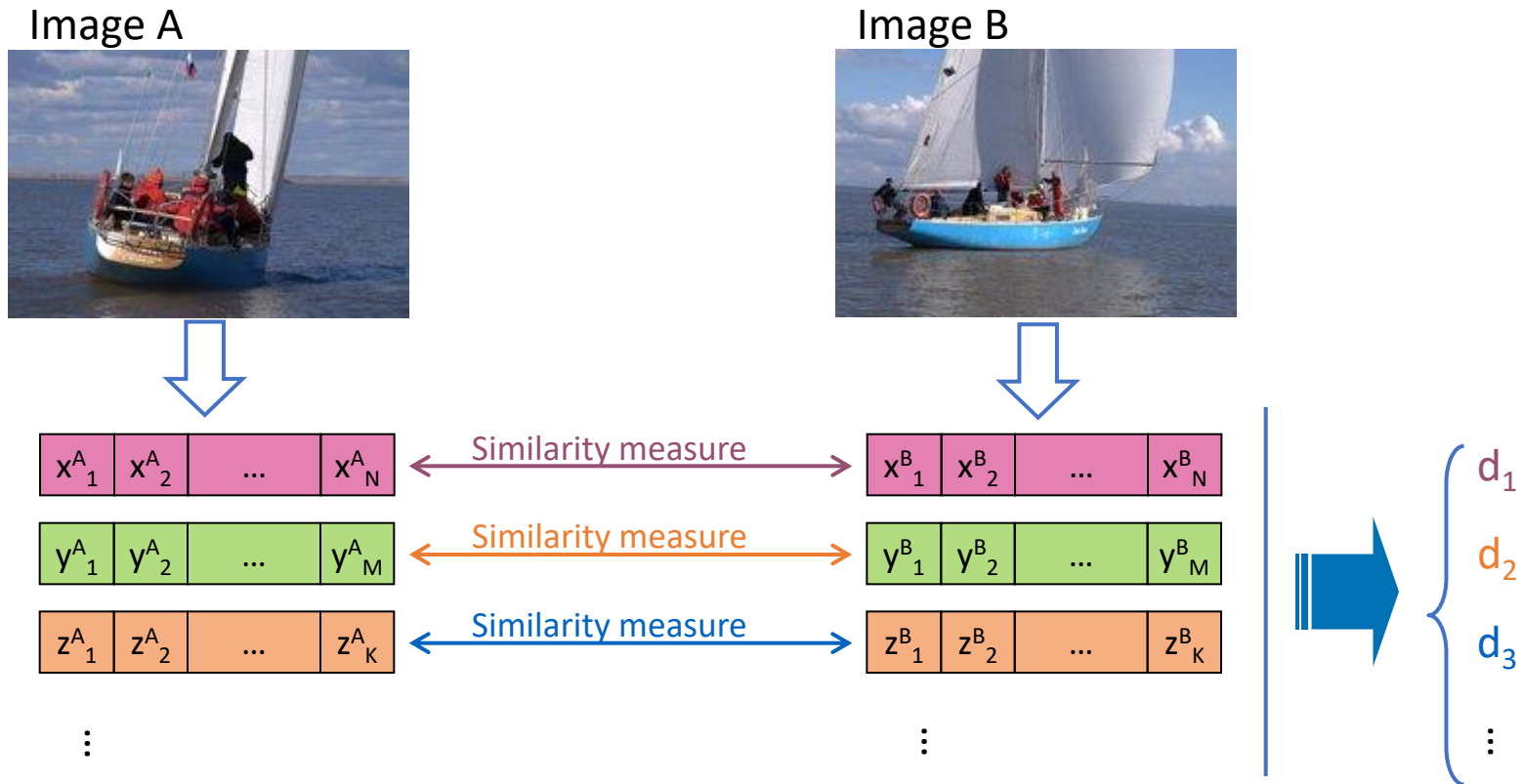


# How to: Feature spaces

- **Feature vector** – a vector of features, representing one image.
- **Feature space** – the set of all possible feature vectors with defined similarity measure.



# How to: Combine results



$$D = \sum_i c_i d_i$$

# Similarity Measure - Manhattan Distance

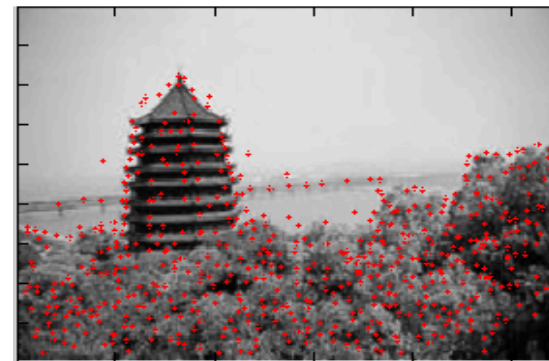
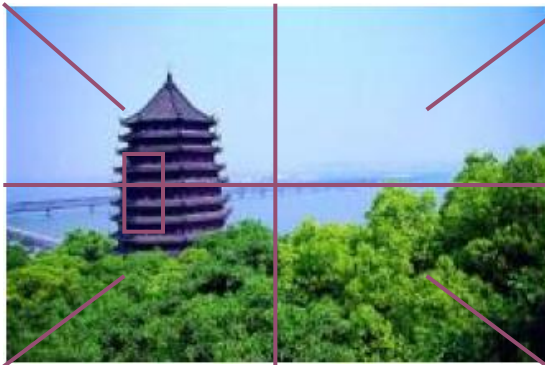
- It is also called  $L_1$  distance. If  $u(x_1, x_2)$  and  $v(y_1, y_2)$  are two points, then the Manhattan distance between  $u$  and  $v$  is given by:
  - $MH(u, v) = |x_1 - y_1| + |x_2 - y_2|$
- Instead of two dimensions, if the points are represented by  $n$ -dimensions, such as  $a = (x_1, x_2, \dots, x_n)$  and  $b = (y_1, y_2, \dots, y_n)$  then the Manhattan distance between  $a$  and  $b$  is calculated as follows:
  - $MH(a, b) = |x_1 - y_1| + |x_2 - y_2| + \dots + |x_n - y_n| = \sum_{i=1}^n |x_i - y_i|$

# Similarity Measure - Euclidean Distance

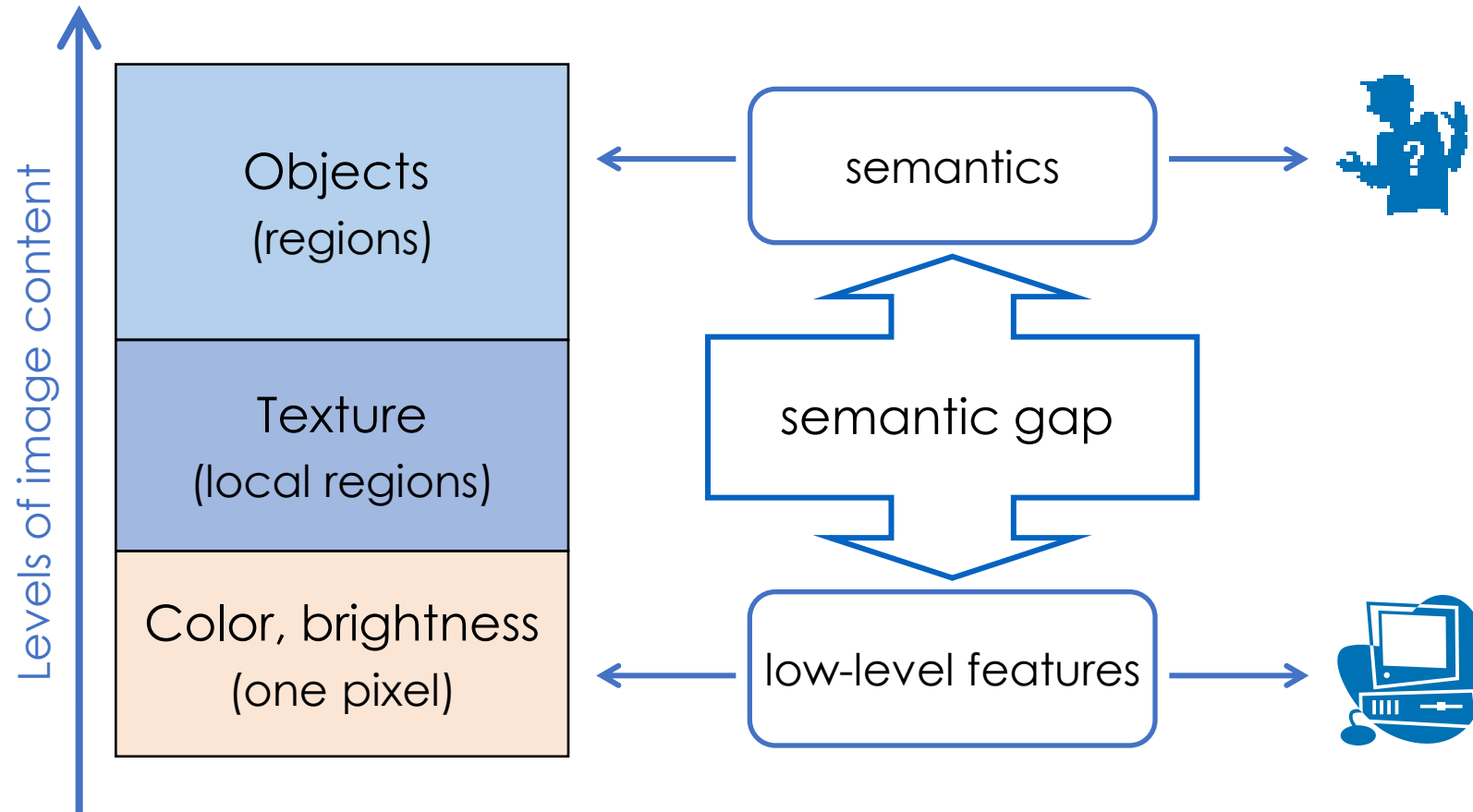
- It is also called  $L_2$  distance. If  $u(x_1, x_2)$  and  $v(y_1, y_2)$  are two points, then the Euclidean distance between  $u$  and  $v$  is given by:
  - $EU(u, v) = \sqrt{|x_1 - y_1|^2 + |x_2 - y_2|^2}$
- Instead of two dimensions, if the points are represented by  $n$ -dimensions, such as  $a = (x_1, x_2, \dots, x_n)$  and  $b = (y_1, y_2, \dots, y_n)$  then the Euclidean distance between  $a$  and  $b$  is calculated as follows:
  - $$EU(a, b) = \sqrt{|x_1 - y_1|^2 + |x_2 - y_2|^2 + \dots + |x_n - y_n|^2}$$
$$= \sqrt{\sum_{i=1}^n |x_i - y_i|^2}$$

# How to: Image segmentation

- Fixed regions
  - The same region boundaries for all images.
- Segmentation
  - Boundaries depends on image content.
- Key points (point of interest) detection
  - Points of particular interest in the image, feature extraction for areas around key points.



# Problems: semantic gap



How to understand what's on the images?

# Problems: what's on the images?



- Sometimes it is not easy to understand the image even for humans!
- What do we want from machines?

# Problems: what's on the images?



- How do we now that all these objects are lamps?



# Lecture Outline

- What is and Why image retrieval?
- How to compare and retrieve images?
  - Common components of the CBIR systems
  - Main problems and research directions
- Performance measurement
  - Retrieval effectiveness

# Performance measurement

## Performance concerns

- Efficiency
  - Important due to the large data size
- Retrieval effectiveness
  - No similarity metric which exactly conforms to human perception

# Problems in effectiveness evaluation

- Define a common image collection
  - Corel Photo CDs
  - Brodatz texture collection: <http://www.ux.uis.no/~tranden/brodatz.html>
  - CoPhIR: <http://cophir.isti.cnr.it/whatis.html>
  - Participate in ImageCLEF, TRECVID, imageEVAL, ROMIP
- Obtain relevance judgement
  - Use of collections with predefined subsets (Corel collection)
  - Image grouping (medical)
  - User judgements
    - Pooling
    - Different types of judgement data (relevant – not relevant, ranking, ...)

# Effectiveness measurement

- “You can see, that our results are better”



# Effectiveness measurement (1)

- Recall and precision

$$precision = \frac{\text{No. relevant documents retrieved}}{\text{Total No. documents retrieved}},$$



$$precision = \frac{true\ positive}{true\ positive + false\ positive}$$

$$recall = \frac{\text{No. relevant documents retrieved}}{\text{Total No. relevant documents in the collection}}$$



$$recall = \frac{true\ positive}{true\ positive + false\ negative}$$

- F-measure

- F1 score is the harmonic mean of precision and recall

$$F1\ Score = 2 * \frac{precision * recall}{precision + recall}$$

# Effectiveness measurement (2)

- Precision at N ( $P@n$ )
  - Precision takes all retrieved documents into account, but it can also be evaluated at a given cut-off rank, considering only the topmost results returned by the system. This measure is called precision at n or  $P@n$ .
- Recall at N
  - Recall at N is the same as the normal recall, but Recall at N is applied into the top-n retrieved image.

# Effectiveness measurement (3)

- Error rate

$$\text{Error rate} = \frac{\text{No. non-relevant images retrieved}}{\text{Total No. images retrieved}}$$

# Effectiveness measurement (4)

- **Mean Average Precision (MAP)**

- The mean average precision which averages precision across all queries  $Q = \{q_1, \dots, q_n\}$ .

- $$MAP = \frac{\sum_{q_i \in Q} \text{precision}(q_i)}{|Q|}$$



# Effectiveness measurement (5)

- **Mean Reciprocal Rank (MRR)**

- the mean reciprocal rank evaluates how good the search mechanism is in ranking the relevant images.
- The main focus of this metric is to find the order of the first relevant image among the retrieved images.

- Assume that  $r_q$  is the rank of the first relevant image for a query  $q$ , then the reciprocal rank of  $q$  is computed as follows.

- $$RR(q) = \begin{cases} 0 & : \text{not relevant} \\ \frac{1}{r_q} & : \text{otherwise} \end{cases}$$

- By considering the multiple participants' responses of each query, the final reciprocal rank per query averages the responses among all users. Thereafter, the mean reciprocal rank for all queries is calculated as follows.

- $$MRR = \frac{1}{|Q|} \sum_{q \in |Q|} RR(q)$$