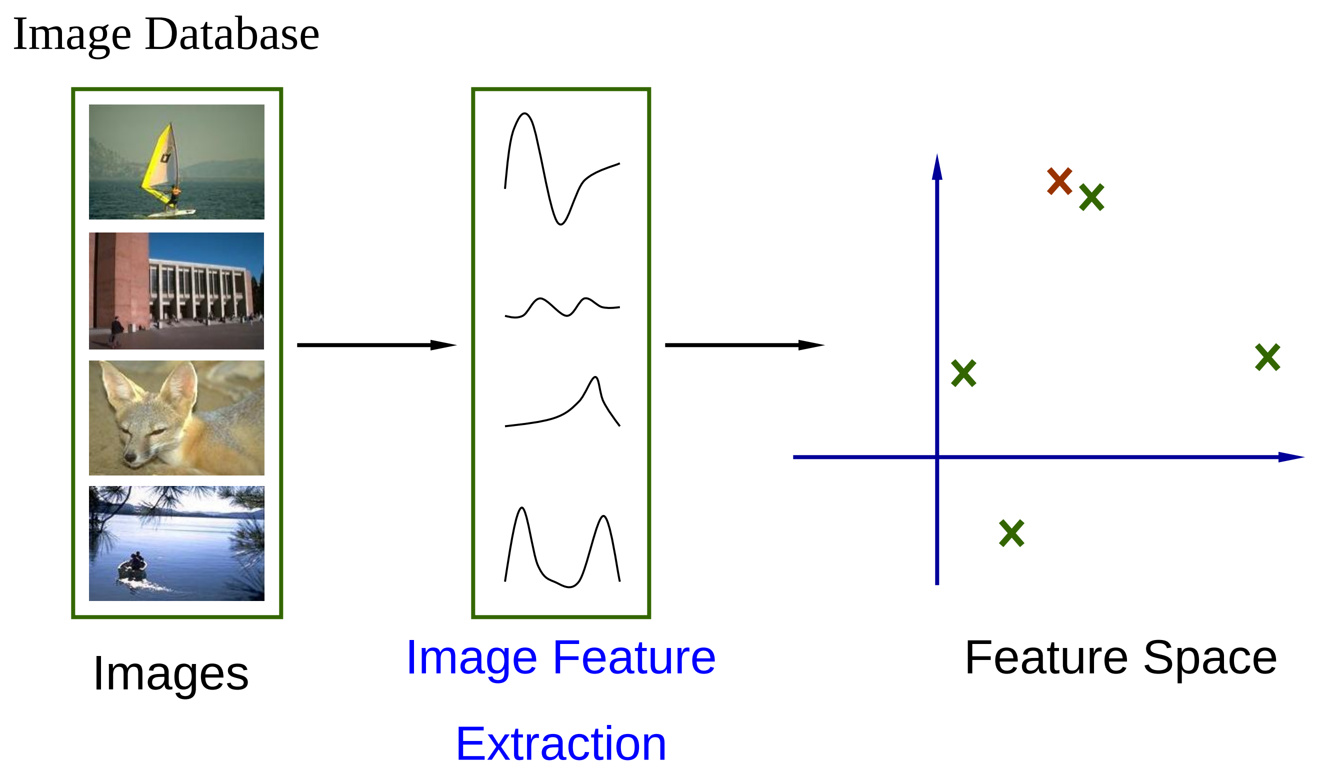
Feature Vectors
Content-based Image Retrieval

Image Features

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

Image Features



Query by Visual Example

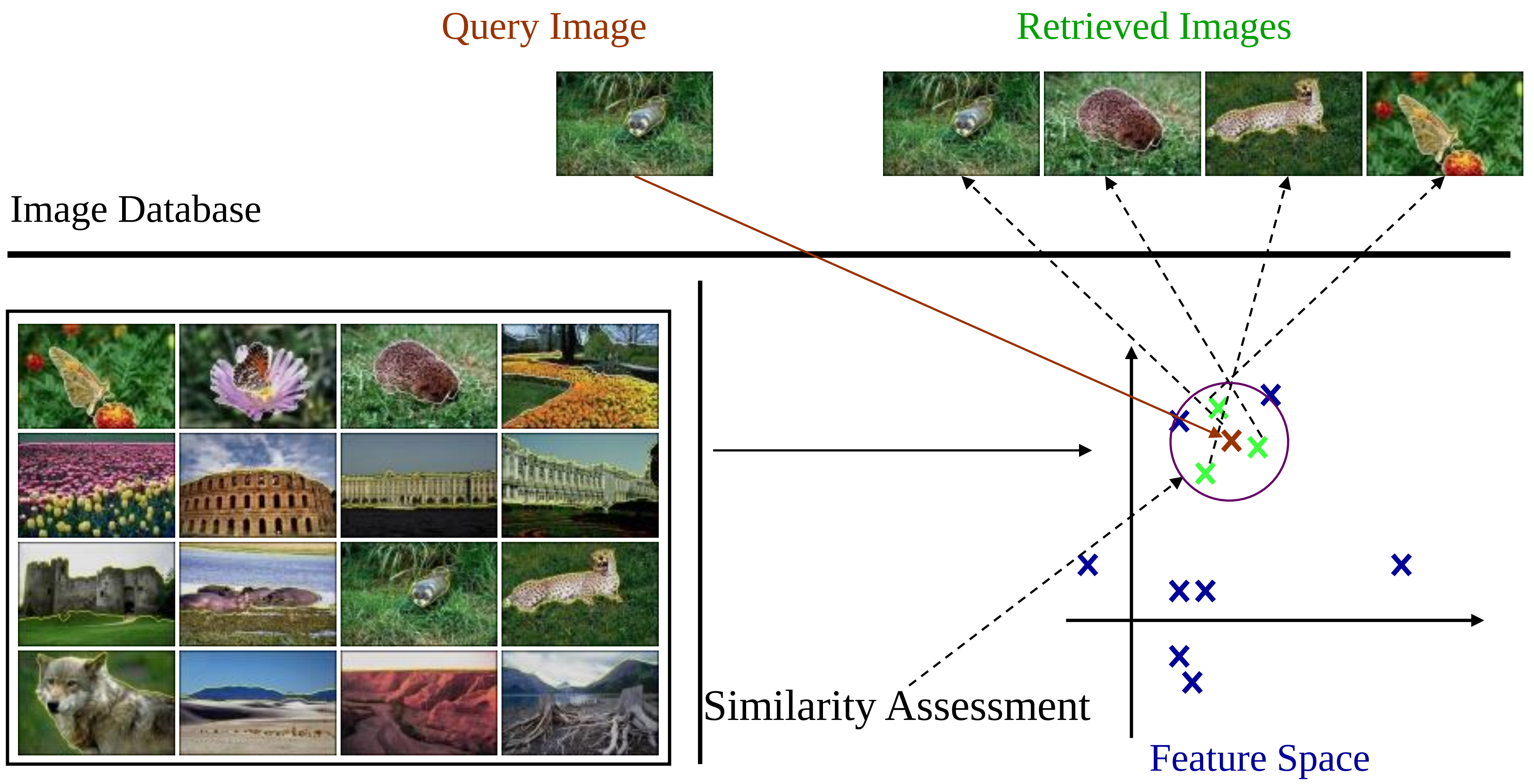


Image Features

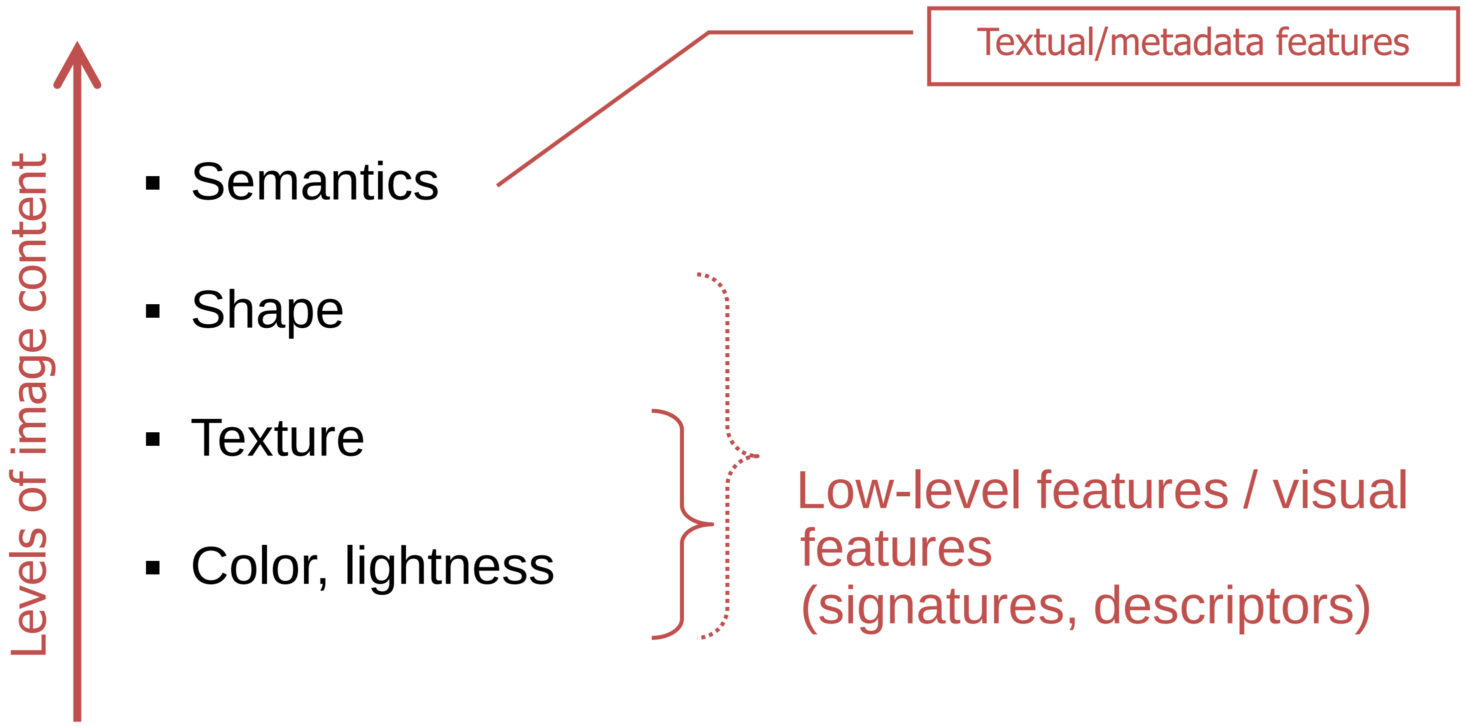
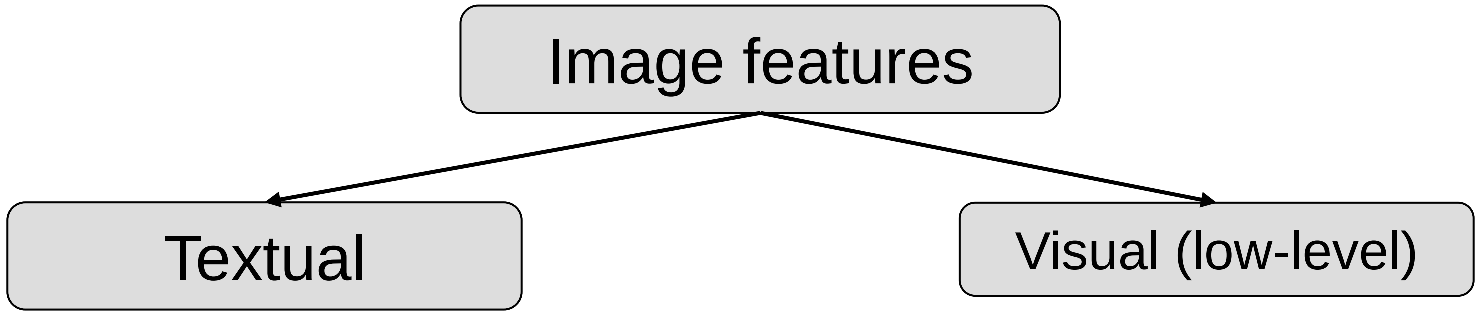

Levels of image content
Semantics
Shape
Texture
Color, lightness
Textual/metadata features
Low-level features / visual features
(signatures, descriptors)

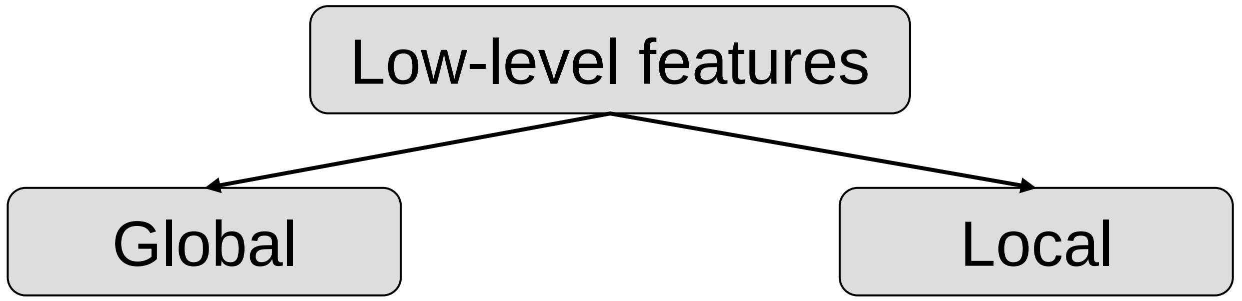
Image Features



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.

Image Features



Describes the whole image:

- average intensity;
- average amount of red;
- …

Describes one part of the image:

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

All pixels of the image are processed.

Segmentation of the image is performed, pixels of a particular segment are processed to extract features, Key point detection and localization

Feature Spaces

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

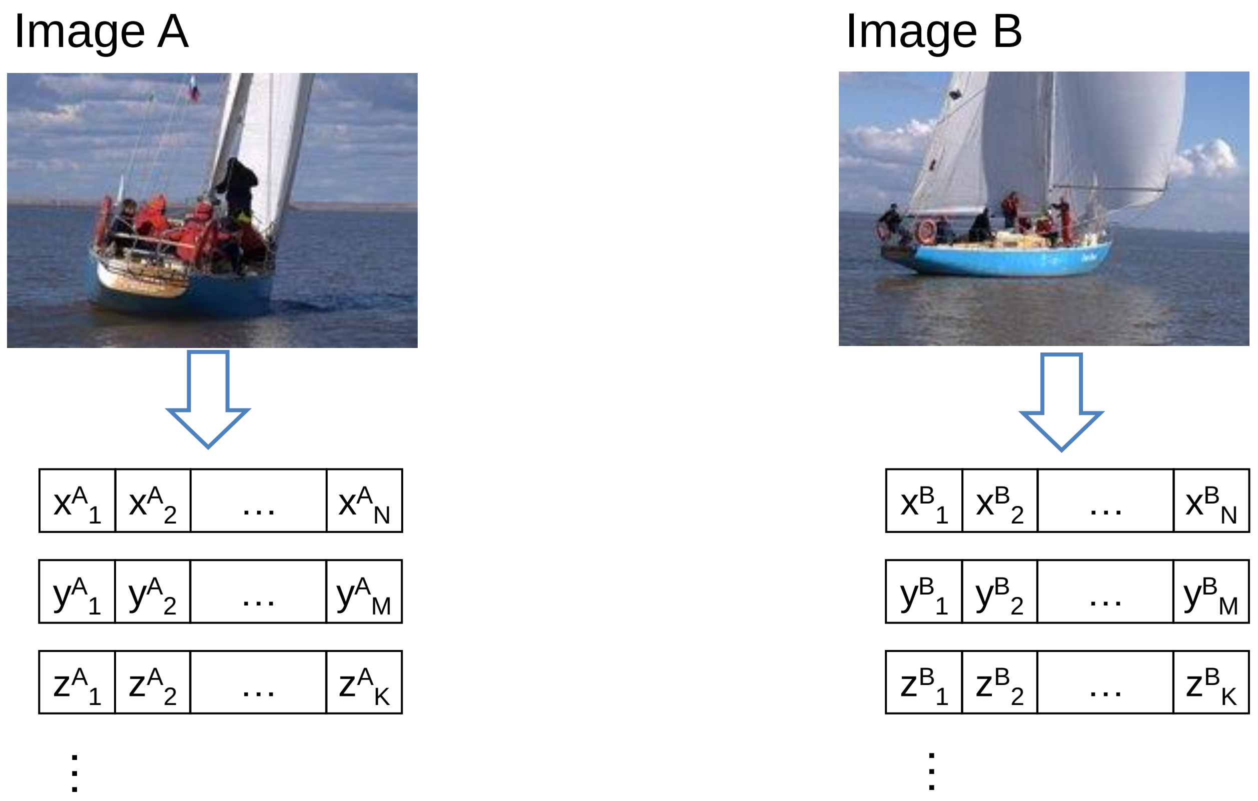


Image A
Image B
x^A_1 x^A_2 … x^A_N
y^A_1 y^A_2 … y^A_M
z^A_1 z^A_2 … z^A_K
⋮
x^B_1 x^B_2 … x^B_N
y^B_1 y^B_2 … y^B_M
z^B_1 z^B_2 … z^B_K
⋮

Feature Spaces

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

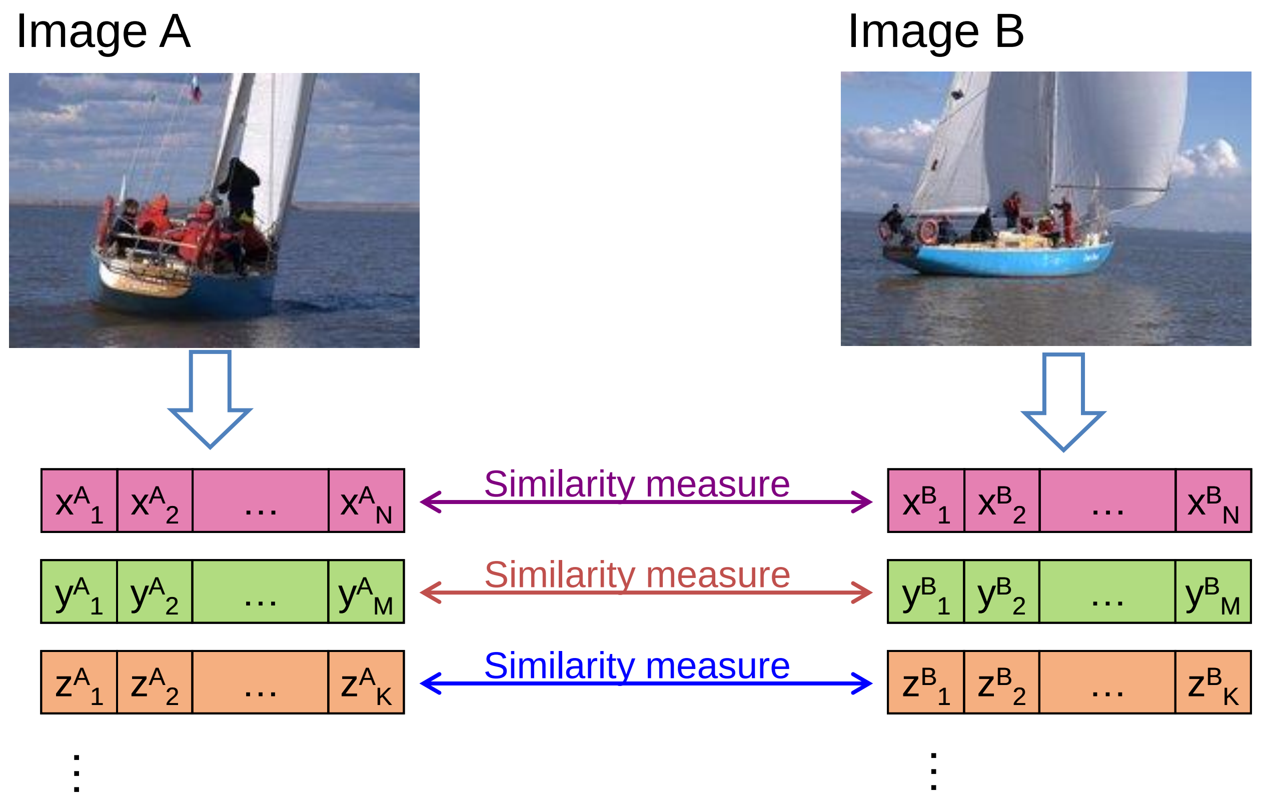


Image A
Image B
x^A_1 x^A_2 … x^A_N
Similarity measure
x^B_1 x^B_2 … x^B_N
y^A_1 y^A_2 … y^A_M
Similarity measure
y^B_1 y^B_2 … y^B_M
z^A_1 z^A_2 … z^A_K
Similarity measure
z^B_1 z^B_2 … z^B_K
⋮
⋮

Feature Spaces

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

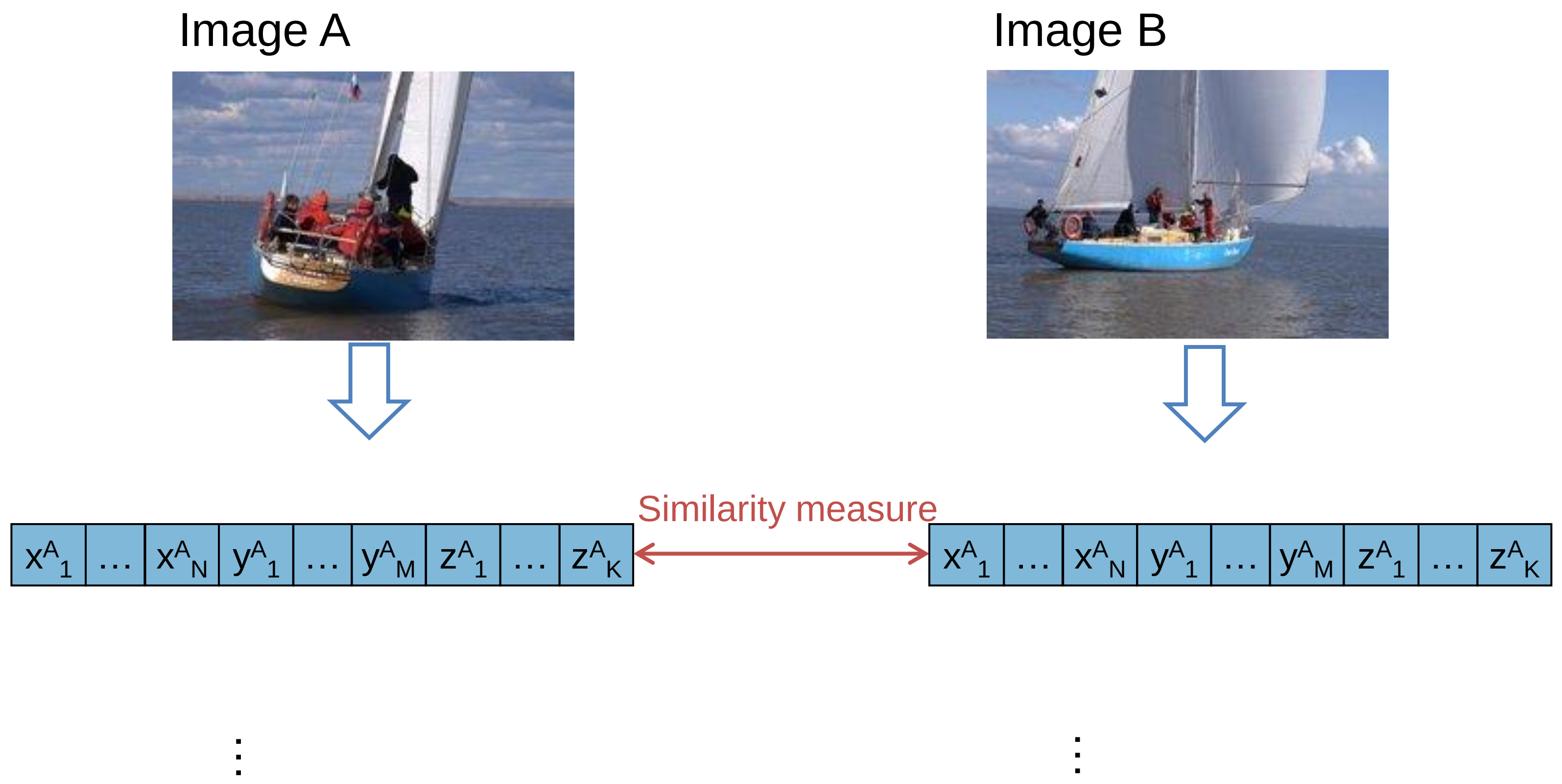


Image A
Image B
Similarity measure
x^A_1 … x^A_N y^A_1 … y^A_M z^A_1 … z^A_K
x^A_1 … x^A_N y^A_1 … y^A_M z^A_1 … z^A_K
⋮
⋮

Combine Results

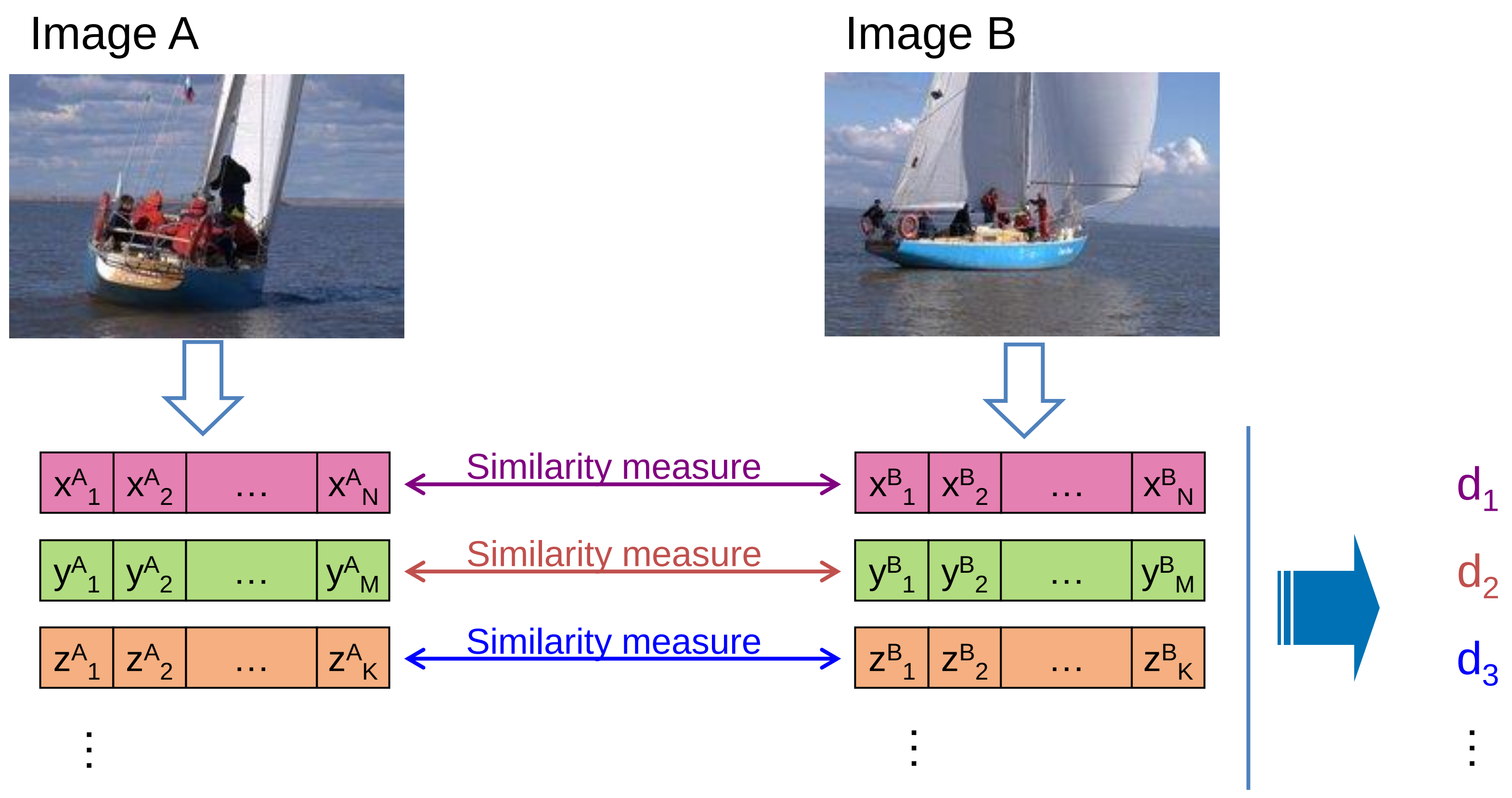


Image A
Image B
x^A_1 x^A_2 … x^A_N
Similarity measure
x^B_1 x^B_2 … x^B_N
y^A_1 y^A_2 … y^A_M
Similarity measure
y^B_1 y^B_2 … y^B_M
z^A_1 z^A_2 … z^A_K
Similarity measure
z^B_1 z^B_2 … z^B_K
d_1
d_2
d_3

$$D = \sum_i c_i d_i$$

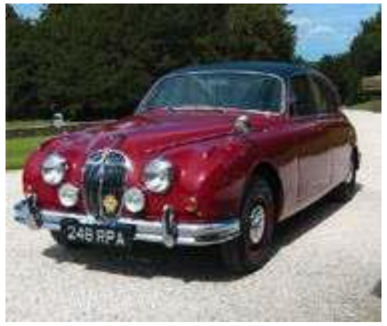
Content-based Image Retrieval

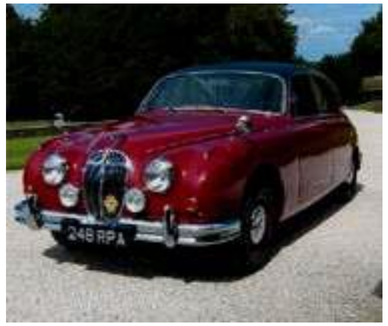
- General: We try to match a query image to an arbitrary collection of images, such as those found on the web. The goal of the query is to obtain images with the same object as the query
  - Given an image with a horse, find all images showing a horse (at least as their main subject)

- Application Specific: We try to match a query image to a collection of images of a specific type.
  - For example, fingerprints, X-ray images of a specific organ, etc.

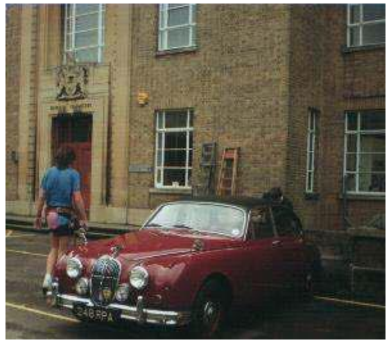
Content-based Image Retrieval

- Query: image/video clip Retrieve: images/shots from archive

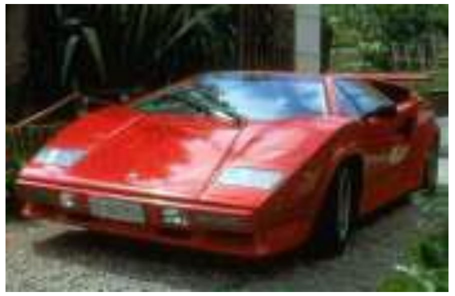




near duplicate



same object



same category

Color Histograms for CBIR

- Histograms can serve as feature vectors (i.e. a list of numbers used to quantify an image and compare it to other images)
- Assumption: images with similar color distributions are semantically similar
- Comparing the "similarity" of color histograms can be done using a distance metric
  - Euclidean, correlation, Chi-squared, intersection, and Bhattacharyya

Using OpenCV to Compute Histograms

- cv2.calcHist(images, channels, mask, histSize, ranges)

- **images**: This is the image that we want to compute a histogram for. Wrap it as a list: [myImage]
- **channels**: A list of indexes, where we specify the index of the channel we want to compute a histogram for. To compute a histogram of a grayscale image, the list would be [0]. To compute a histogram for all three red, green, and blue channels, the channels list would be [0, 1, 2]
- **mask**: uint8 image with the same shape as our original image, where pixels with a value of zero are ignored and pixels with a value greater than zero are included in the histogram computation. Using masks allow us to only compute a histogram for a particular region of an image
- **histSize**: This is the number of bins we want to use when computing a histogram. Again, this is a list, one for each channel we are computing a histogram for. The bin sizes do not all have to be the same. Here is an example of 32 bins for each channel: [32, 32, 32]
- **ranges**: The range of possible pixel values. Normally, this is [0, 256] for each channel, but if you are using a color space other than RGB (such as HSV), the ranges might be different

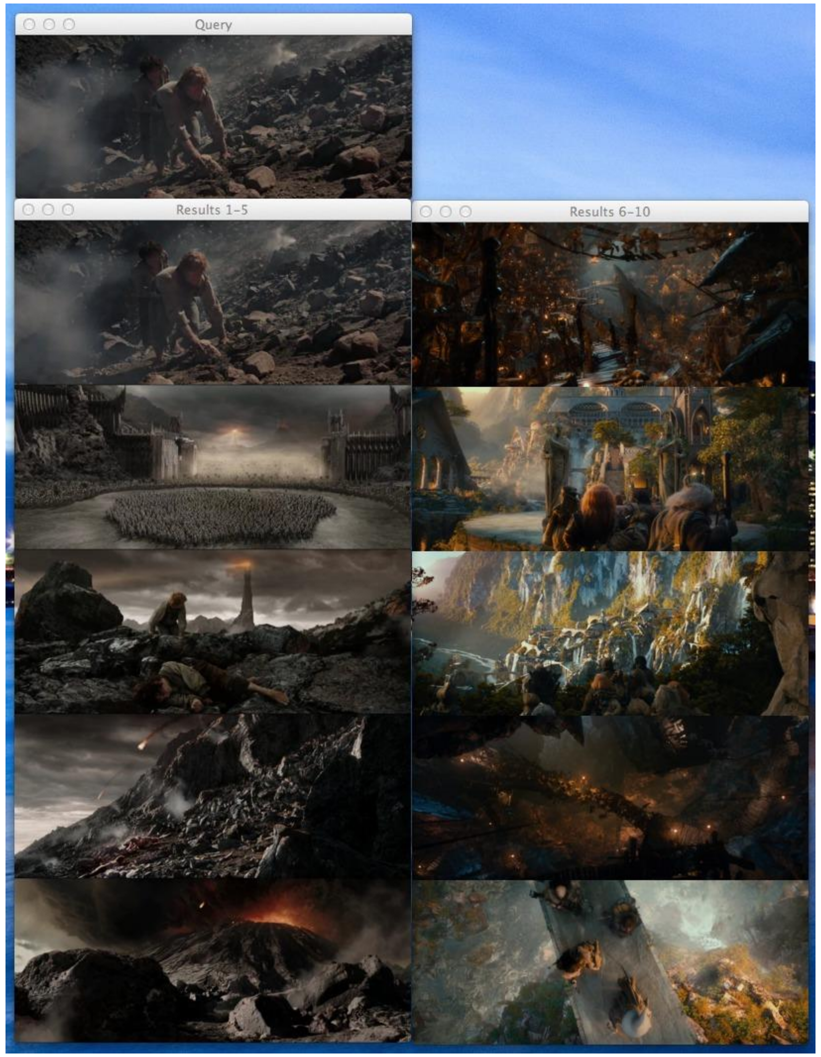
Compare Histograms

- cv2.compareHist(H1, H2, method)
  - cv2.cv.CV_COMP_CORREL: Computes the correlation between the two histograms
  - cv2.cv.CV_COMP_CHISQR: Applies the Chi-Squared distance to the histograms

$$\frac{1}{2}\sum_{i=1}^{n}\frac{(x_i - y_i)^2}{(x_i + y_i)}$$

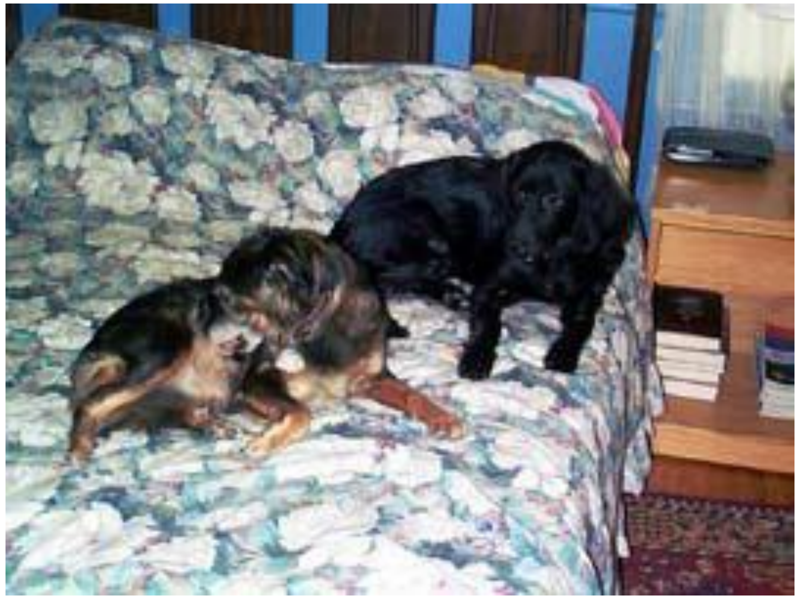
  - cv2.cv.CV_COMP_INTERSECT: Calculates the intersection between two histograms
  - cv2.cv.CV_COMP_BHATTACHARYYA: Bhattacharyya distance, used to measure the “overlap” between the two histograms
  - cv2.cv.CV_COMP_HELLINGER: A synonym for cv2.cv.CV_COMP_BHATTACHARYYA

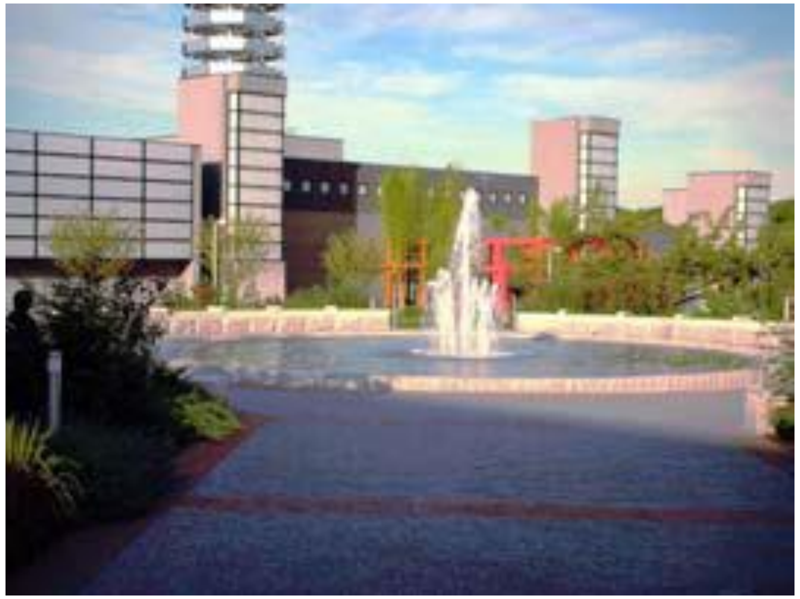
Example: Query by Color



Problems

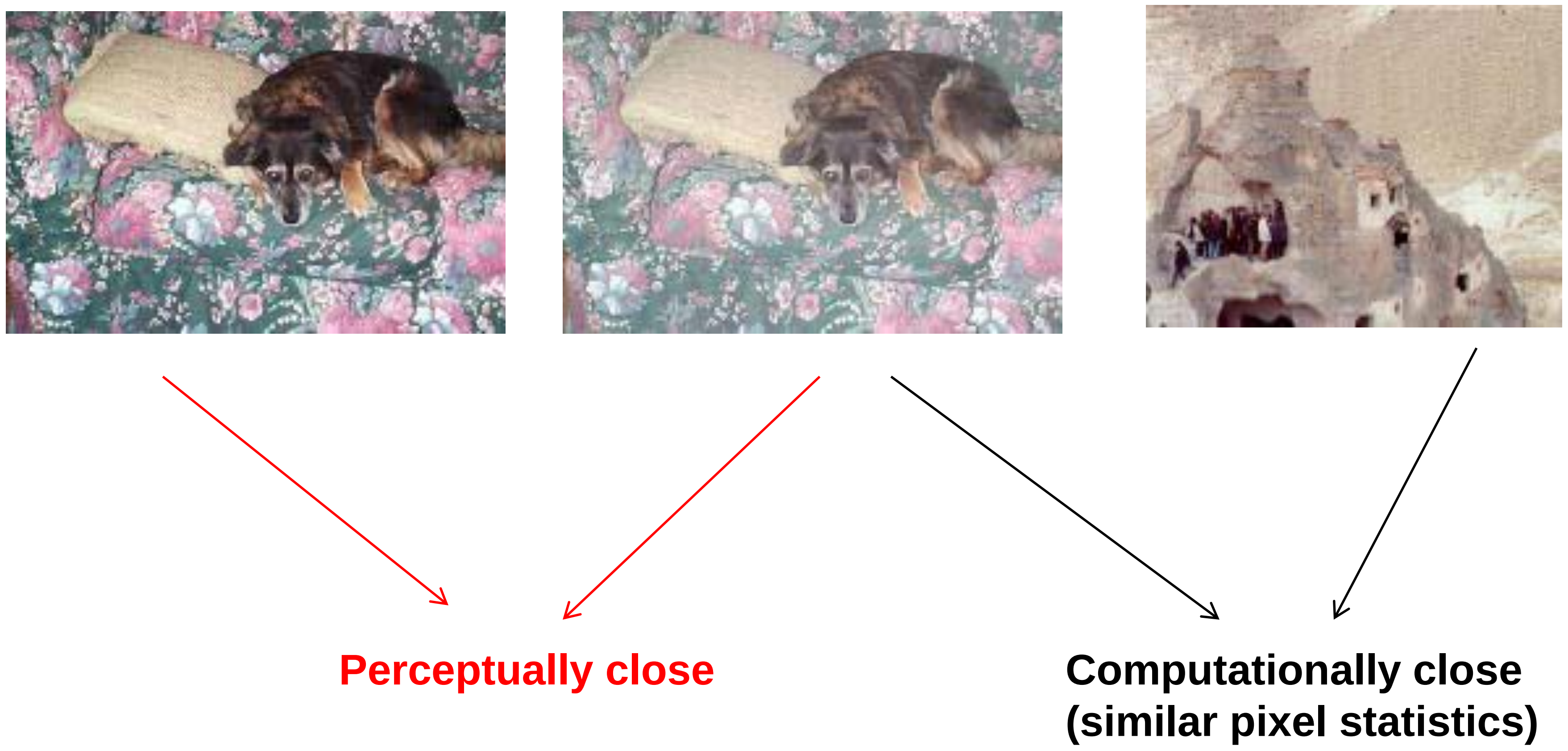
- Two Images with nearly identical color histograms





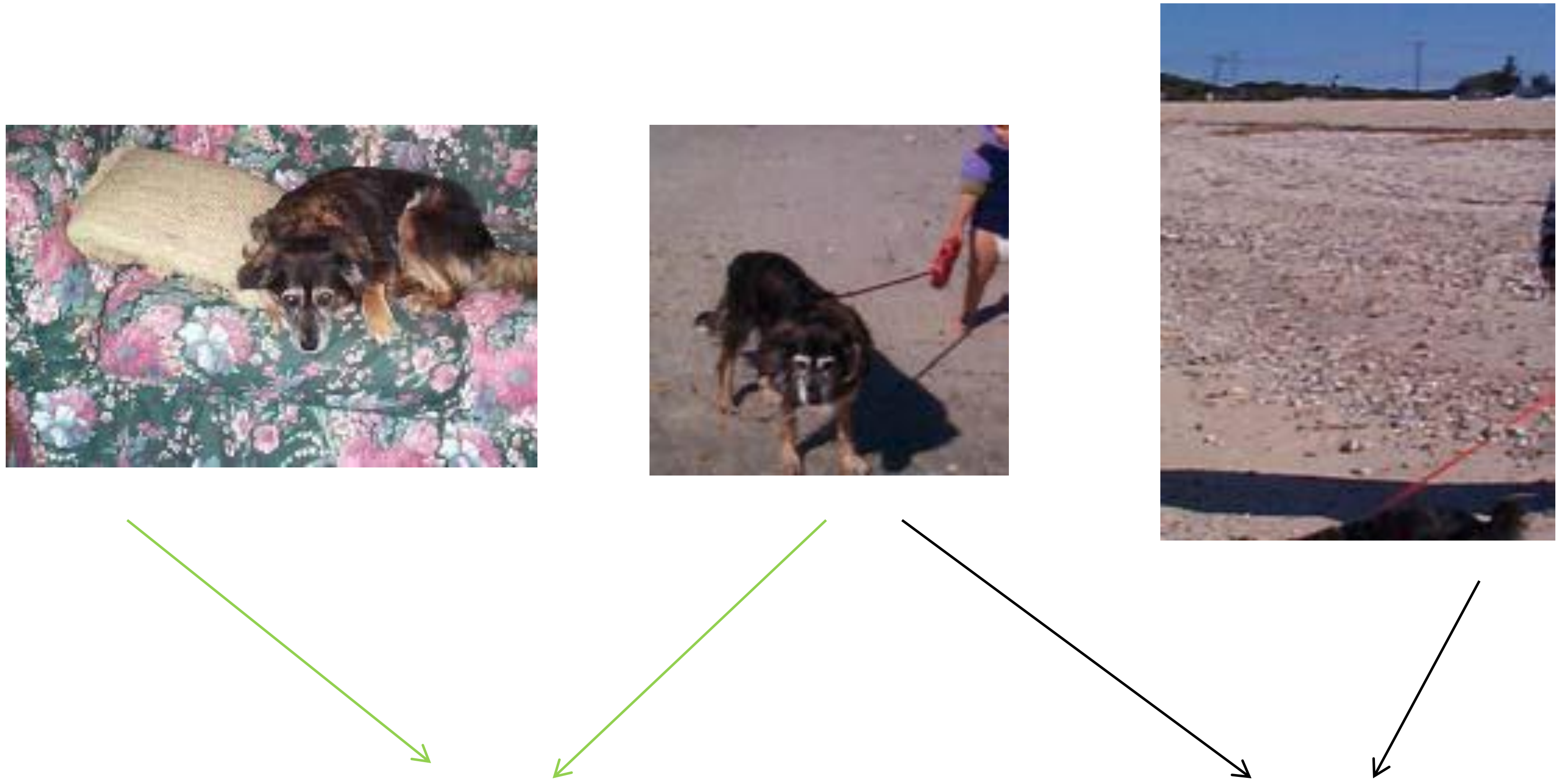
Problems

- Perceptual versus computational similarity



Problems

- Conceptual similarity is even harder to deal with

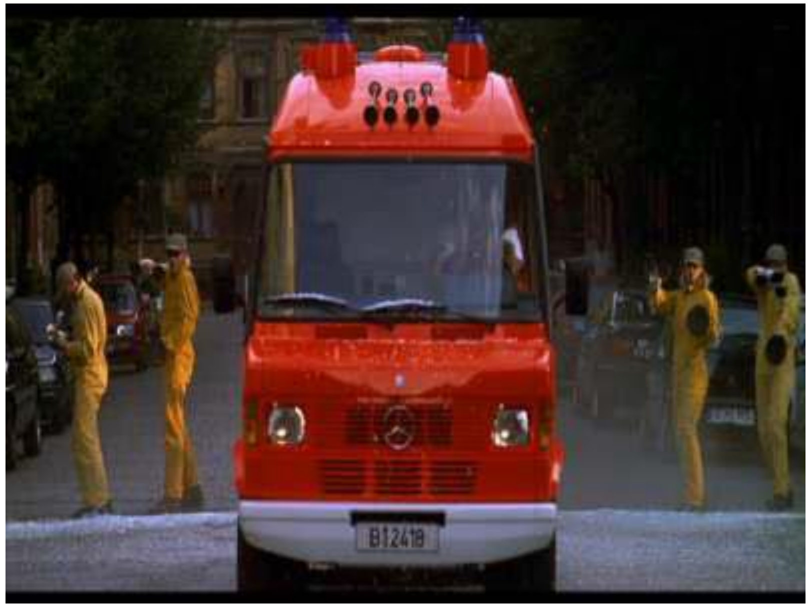
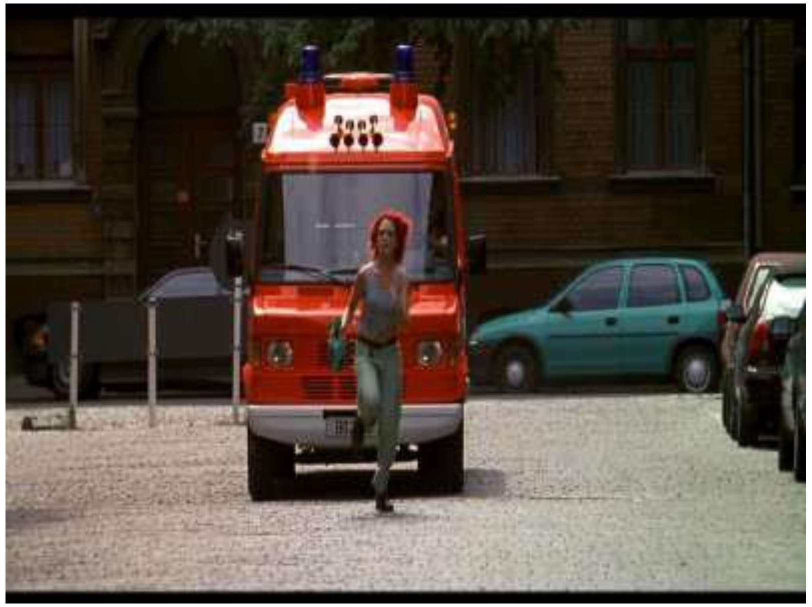


Conceptually Close

Computationally close
(Large areas with similar local pixel statistics)

Problems

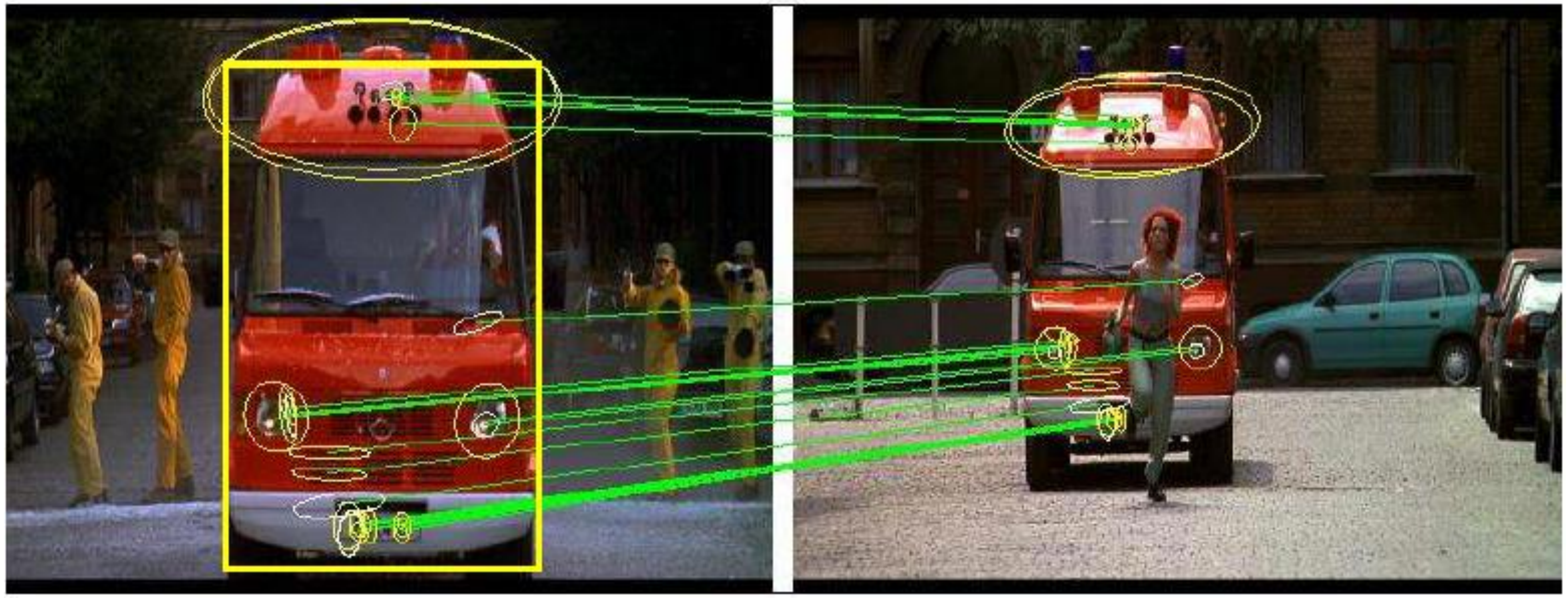
- Particular objects, not entire images

- Forced to face problems of:
  - scale change
  - pose change
  - illumination change
  - partial occlusion

Possible solution

- When do (images of) objects match?



- Two requirements:
  - “patches” (parts) correspond, and
  - Configuration (spatial layout) corresponds