Object Recognition

Chapter 3: Global Features

Prof. Dr. Johannes Maucher

HdM CSM

Version 1.1
27.03. 2019

Document History

Version Nr.	Date	Changes
1.0	18.02.2013	Initial Version
1.1	27.03.2019	Adaptations for SS 19

Chapter 3: Global Features

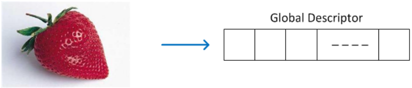
Appearance Based Recognition and Global Features

Appearance-Based Recognition

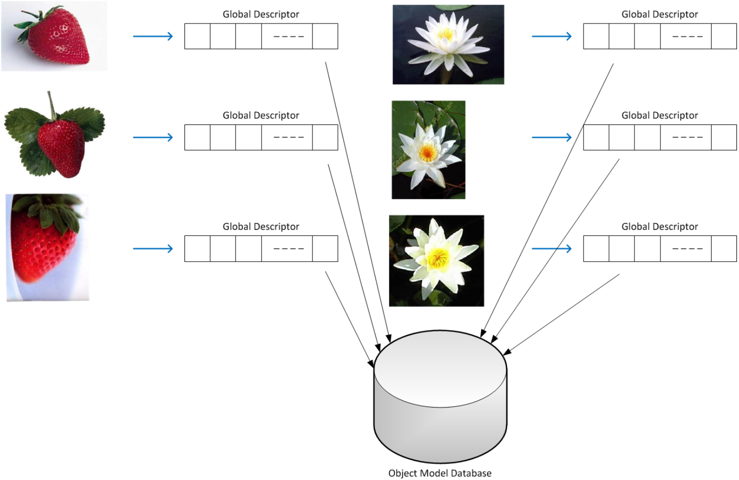
- Objects can be represented by a set of 2D images (*appearances*)
- Recognition: Compare the 2D appearances

Global Feature

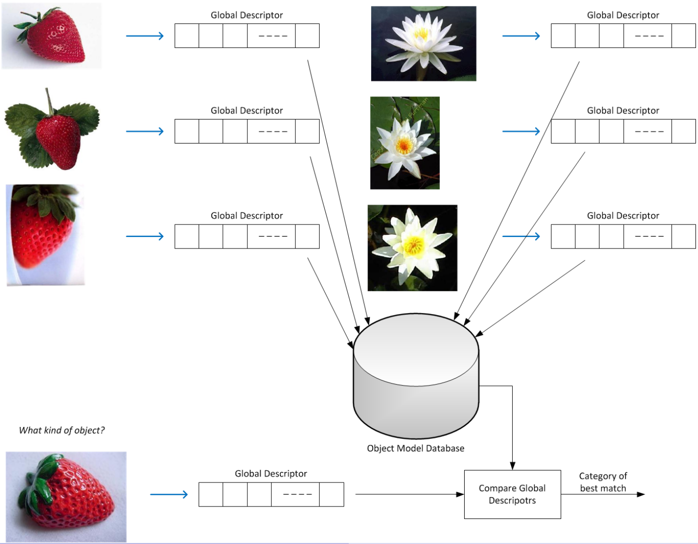
A single feature descriptor is calculated for the entire image or cropped object. I.e. the appearance is encoded in a single feature descriptor.



Calculate Model



Object Recognition



Pixel Intensities

- Given: L-channel image of r rows and c pixels per row:
- Feature Descriptor: Spatially ordered vector of $L \cdot r \cdot c$ pixel intensity values.
- Pixel intensities are either integers in the range of $\left[0, \ldots, 2^b - 1\right]$ or their normalized representations, i.e. 2^b discrete values in the range $[0, \ldots, 1]$, where b is the number of bits per pixel and channel.

Example: Pixel Intensities

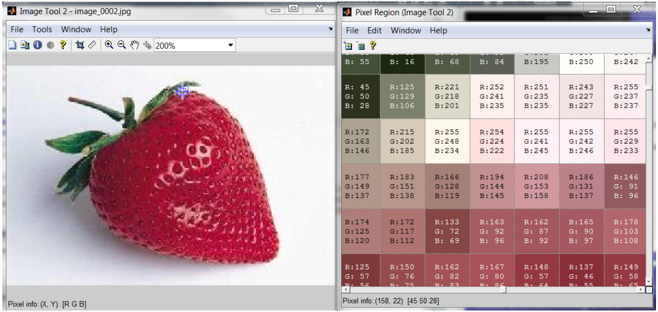


Abbildung: RGB image and pixel intensities of the subregion, indicated by the blue rectangle in the image.

- RGB image of size (200*x*300), 8 Bit per channel and pixel
- Length of feature descriptor:

$$L = 3 \cdot 300 \cdot 200 = 180000$$

Requirements and Drawbacks

- Images must be cropped to the object of interest
- Objects must be aligned in terms of orientation and pose
- (All objects of a category should have similar colors) $\Leftarrow$ Actually this approach is not suitable for categorization.
- Suitable, e.g. for cropped faces
- Extremely high-dimensional features space
- Not robust w.r.t. translations, rotations, scale, view point, pose, illumination.

1-dimensional histograms per channel

Abbildung: RGB-Image Alcatraz

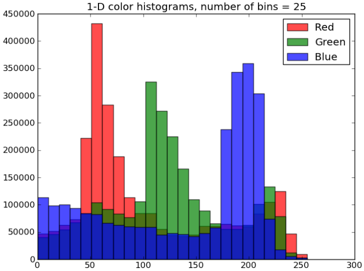


Abbildung: Histograms of R- G- and B-channel of the Alcatraz Picture

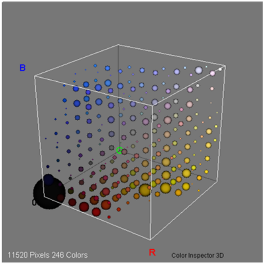
3-Dimensional Color Histogram

- The intensities of each pixel in a RGB-Image[1] can be represented as a point in a 3-Dimensional space.
- If b is the number of bits per pixel and channel, then there exist $z = 2^{3b}$ different points in the 3-dimensional RGB space.
- Partition the space of z points into $n < z$ 3-dimensional bins.
- The 3-dimensional color histogram counts for each bin how often points within the bin occur in the image.
- Apply the 3D color histogram as global feature descriptor. The corresponding feature vector has length n (number of bins).
- For grayscale images the histogram is 1-dimensional.
- Introduced by Swain and Ballard in [4]. There the almost error-free recognition of 66 objects has been demonstrated.

[1] In this section we assume RGB. However, the concept of color histograms can also be applied in other color spaces

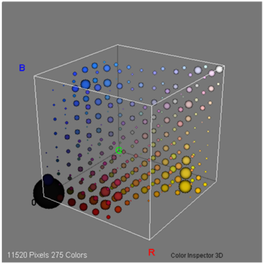
Example of 3-dimensional Color Histogram

Image



3D Color histogram of image

Rotated Image



3D Color histogram of rotated image

Source: [4]

Serialisation of 3-dimensional Histogram

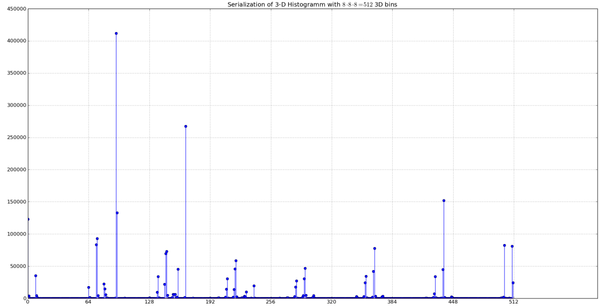
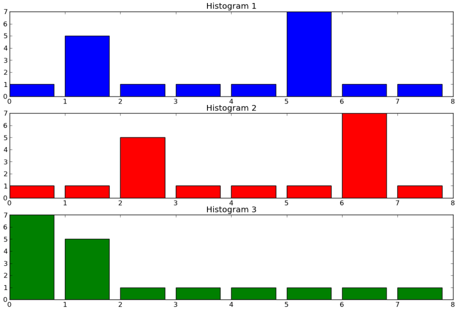


Abbildung: Serialization of the 3D histogram with $8 \cdot 8 \cdot 8 = 512$ bins (Alcatraz image).

Characteristics of Color Histogram

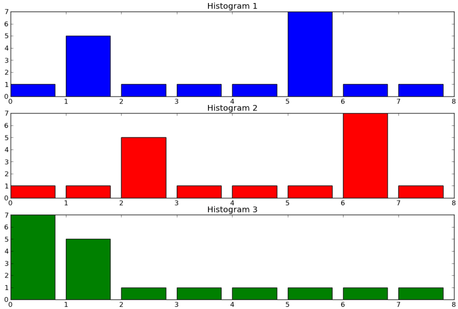
- In contrast to pixel intensity descriptor the histogram is orderless, i.e. it does not matter where the color occurs.
- Invariant w.r.t. geometric transformations
- Robust to view point
- Robust to partial occlusion
- Robust w.r.t. to contrast change (if intensity-normalized values are used)
- Not robust w.r.t. illumination changes
- Not suitable if objects of same class can have different colors.

How to compare histograms?



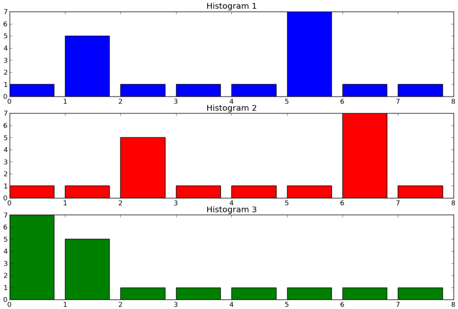
- Question: Is histogram 2 or histogram 3 closer to histogram 1?

How to compare histograms?



- Question: Is histogram 2 or histogram 3 closer to histogram 1?
- Answer: Depends on method to measure similarity or distance.

How to compare histograms?



- Question: Is histogram 2 or histogram 3 closer to histogram 1?
- Answer: Depends on method to measure similarity or distance.
- Distance is the inverse of similarity.

Normalization

- Notation: H_q is a histogram and $H_q(i)$ is the count in the $i.th$ bin of H_q
- Normalization: Histograms shall be normalized before they are compared. Normalized histogram

$$H_q := \frac{1}{m_q} \left(H_q(0), H_q(1), H_q(2), \ldots, H_q(Z) \right)$$

where

$$m_q = \sum_{i=1}^{Z} H_q(i)$$

Euclidean Distance

- Euclidean Distance:

$$d_E(H_q, H_p) = \sqrt{\sum_i (H_q(i) - H_p(i))^2} \tag{1}$$

- Characteristics:
  - All bins are weighted equally
  - Strong impact of outliers
  - Bin-by-Bin comparison. Drawback: If two images are identical up to a small brightness change, then the histograms look similar up to a small shift along the x-axis. However, such a type of similarity can not be measured by a Bin-by-Bin metric.

Pearson Correlation Distance

- Pearson Correlation Distance:

$$d_P(H_q, H_p) = 1 - \frac{\sum_i (H_q(i) - \mu_q) \cdot (H_p(i) - \mu_p)}{\sqrt{\sum_i (H_q(i) - \mu_q)^2 \cdot \sum_i (H_p(i) - \mu_p)^2}} \qquad (2)$$

  where

$$\mu_q = \frac{m_q}{Z}$$

  is the mean over all bins of H_q.
- Characteristics:
  - Bin-by-bin comparison
  - Ignores different offsets (not relevant in the case of normalized inputs) and is an (inverse) measure for linearity

Bray-Curtis Distance

- Bray-Curtis Distance:

$$d_{BC}(H_q, H_p) = \frac{\sum_i |H_q(i) - H_p(i)|}{\sum_i |H_q(i) + H_p(i)|} \tag{3}$$

- Characteristics:
  - Bin-by-bin comparison
  - More robust than euclidean distance w.r.t. outliers
  - Each bin difference is weighted equally

Canberra Distance

- Canberra Distance:

$$d_{Ca}(H_q, H_p) = \sum_i \frac{|H_q(i) - H_p(i)|}{|H_q(i)| + |H_p(i)|} \tag{4}$$

- Characteristics:
  - Bin-by-bin comparison
  - Compared to Bray-Curtis now each bin difference is weighted individually.

Bhattacharyya Distance

- Bhattacharyya Distance:

$$d_{Ba}(H_q, H_p) = \sqrt{1 - \sum_i \sqrt{H_q(i) \cdot H_p(i)}} \tag{5}$$

- Characteristics:
  - Bin-by-bin comparison
  - Requires normalized inputs
  - Statistically motivated for measuring similarity between probability distributions. Here applied for the univariate case, but also applicable for multivariate distributions.

Chi-Square Distance

- Chi-Square Distance (χ^2):

$$d_{\chi^2}(H_q, H_p) = \sum_i \frac{(H_q(i) - H_p(i))^2}{H_q(i) + H_p(i)} \tag{6}$$

- Characteristics:
    - Bin-by-bin comparison
    - Each bin is weighted individually
    - Statistically motivated for measuring similarity between probability distributions. Here applied for the univariate case, but also applicable for multivariate distributions.

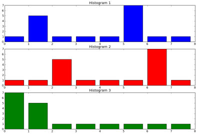
Intersection Distance

- Intersection Distance:

$$d_{IS}(H_q, H_p) = \sum_i \min(H_q(i), H_p(i)) \tag{7}$$

- Characteristics:
  - Requires normalized inputs
  - Bin-by-bin comparison
  - Measures intersection of both histograms
  - Introduced in [4] for color histogram comparison

Example Distance Measures




```
----- Normalized Histograms -----
H1 =  [ 0.06  0.28  0.06  0.06  0.06  0.39  0.06  0.06]
H2 =  [ 0.06  0.06  0.28  0.06  0.06  0.06  0.39  0.06]
H3 =  [ 0.39  0.28  0.06  0.06  0.06  0.06  0.06  0.06]
```

----- Histogram Comparison -----

Method	d(H1,H2)	d(H1,H3)	d(H2,H3)
euclidean	0.566557723733	0.471404520791	0.566557723733
correlation	1.3164556962	0.911392405063	1.3164556962
canberra	2.83333333333	1.5	2.83333333333
braycurtis	0.555555555556	0.333333333333	0.555555555556
mindist	0.692307692308	0.6	0.692307692308
bhattacharyya	0.485132365836	0.387907304067	0.485132365836
chi2	0.796296296296	0.5	0.796296296296

Histogram Comparison: Which is the best?

- All of the distances, introduced so far, compare bins and therefore ignore shift-similarity
- Which distance method is the best depends on the application.
- Common Procedere: Test different distance methods in your application and determine the best.
- Example: See following slides

Example Image Retrieval

Goal: For a given query-image search in an image database the most similar images. Or better: Find images in database, which contain the same objects as the query-image.

Nearest Neighbor Histogram based Image Recognition Algorithm

1. Generate database of histograms $H = \{H_i\}$: For each view of each object one histogram.
2. Calculate Histogram H_t of query-image.
3. Compare H_t to each $H_i \in H$, using an appropriate distance method.
4. Retrieve the image, whose histogram is closest to H_t, or reject query if no close histogram exists.

Image Retrieval Example: Part of 66 Objects Image Database

2

[2] Source: http://www.vision.ee.ethz.ch/datasets/index.en.html

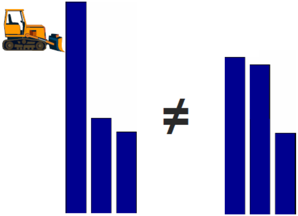
Image Retrieval Example - Results

- Each of the 266 images has been used once as query
- Correct: If the found image contains the same object as the query
- Error: Otherwise

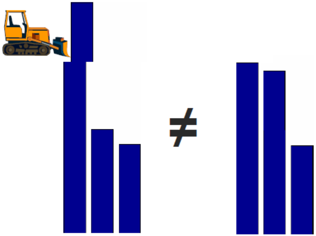
Method	Correct	Error
euclidean	154	111
correlation	151	114
canberra	254	11
braycurtis	229	36
intersection	229	36
bhattacharyya	247	18
chi2	240	25

EMD (see next slides) with 25*x*25 bins: 234 correct, 32 error

Earth Mover Distance [1]



Earth Mover Distance



Earth Mover Distance

$$Work = \sum (\text{ distance moved }) \cdot (\text{ amount moved })$$

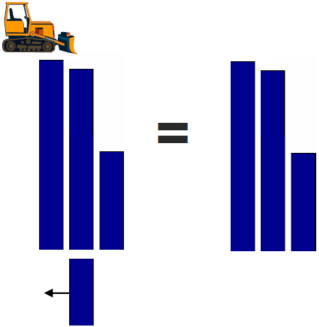
$$Work = \sum_{a=1}^{m} \sum_{b=1}^{n} d_{a,b} \cdot f_{a,b} \tag{8}$$

where m is the number of bins in histogram H_i and n is the number of bins in histogram H_j.

- Earth Mover Distance $d_{emd}(H_i, H_j)$ between histogram H_i and histogram H_j is the minimum possible amount of work to convert H_i into H_j, divided by the sum over all $f_{a,b}$:

$$Work = \frac{\sum_{a=1}^{m} \sum_{b=1}^{n} d_{a,b} \cdot f_{a,b}}{\sum_{a=1}^{m} \sum_{b=1}^{n} f_{a,b}} \tag{9}$$

- EMD regards distance between similar bins and is not just a bin by bin comparison.
- EMD is expensive to calculate



Pros and Cons of Color Histograms

Advantages

- Invariant to object translation and rotation
- Robust w.r.t. deformable objects (e.g. clothes)
- Gradually change of histogram when parts of objects are occluded.
- Slow histogram change in case of out-of-plane rotation
- Doesn't require perfect segmentation

Disadvantages

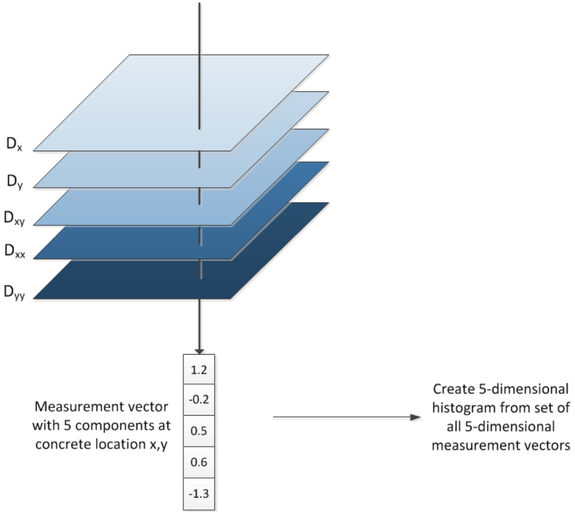
- Strong change with illumination
- Not all objects can be identified by their color distribution (e.g. lemon, banana)

Extension of Color Histogram Descriptor to multidimensional receptive fields

- Introduced by Schiele and Crowley in [2] and extended in [3].
- Idea: Include not only color- but also shape-information into the histogram.
- Shape information can be extracted by applying derivative filters.
- At each location (x, y) in the image calculate the response to a set of K filters.
- Each of these filters constitutes a local receptive field.
- At each location (x, y) the K response-values constitute a $K-$dimensional measurement vector. Usually this vector is extended by C color values[3], yielding a $(K + C)$-dimensional measurement vector.
- All $(K + C)$-dimensional measurement vectors constitute a $(K + C)$-dimensional histogram, which can be applied for object recognition in the same way as described for color histograms.

[3] C is usually 3 for RGB images, 2 for HSV color channels or 1 for greyscale images

Example: Create 5-dimensional histogram from derivatives



Commonly applied descriptors

Gradient D_x, D_y: Rotation variant; Can be applied to detect oriented structures, e.g. vertical lines.

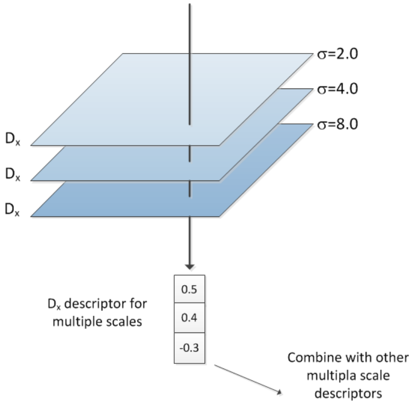
Direction of Gradient $Dir = \arctan \frac{D_y}{D_x}$: Rotation variant; Can be applied to detect oriented structures, e.g. vertical lines.

Magnitude of Gradient $Mag = \sqrt{D_x^2 + D_y^2}$: Rotation invariant.

Laplacian $Lap = D_{xx} + D_{yy}$: Rotation invariant.

Multiscale Descriptors

- Experiments have shown that simple combinations, e.g. (Magnitude,Laplacian) are sufficient.
- But it is important to calculate the descriptor at different scales
- Can be easily implemented by applying derivatives of Gaussian filters.
- For example a (Magnitude,Laplacian)-Descriptor, calculated at 3 different scales yields a 6-dimensional measurement vector.



Drawback of Nearest Neighbor Recognition Algorithm

- For global histogram features, either color histogram or the extension to multidimensional receptive fields, the Nearest Neighbor Matching Algorithm, sketched on page 26 can be applied.
- Drawbacks of Nearest Neighbor Matching Algorithm:
  - In the recognition phase each histogram in the database must be compared to the histogram of the query-image.
  - Storage of multi-dimensional histograms quite expensive
  - Not robust in the case of occluded objects.
- Nearest Neighbor Strategies are lazy learners. I.e. they do not learn a model, but keep all data.
- On the other hand there exists lots of machine learning algorithms, e.g. The Naive Bayes Classifier, that learn a model, which is then applied in the recognition phase.

Probabilistic Object Recognition according to [3]

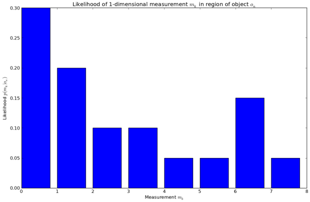
Training:

- For each of the N objects o_n in the database, calculate the Likelihood

$$p(m_k|o_n)$$

that the vector m_k is measured in the region of this object.

- The Likelihood $p(m_k|o_n)$ can be easily obtained from the multidimensional histogram of object o_n (see previous sections), if the histograms are normalized to an integral value of 1.
- Example for 1-dimensional histogram:



Probabilistic Object Recognition

Recognition

- Take a set M of measurements in the query-image

$$M = (m_1 \wedge m_2 \wedge m_3 \wedge \cdots m_z)$$

- Apply Bayes Theorem in order to calculate the A-posteriori probability that given measurements M the corresponding object is o_n

$$p(o_n|M) = \frac{p(M|o_n) \cdot p(o_n)}{p(M)} = \frac{p(M|o_n) \cdot p(o_n)}{\sum_i^N p(M|o_i) \cdot p(o_i)} \qquad (10)$$

- Decision: For given measurements M calculate $p(o_n|M)$ for all possible objects and decide for o_n, which yields maximum $p(o_n|M)$.

Probabilistic Object Recognition

Recognition:

- Calculation of A-Posteriori (Equation (10)):
  - $p(o_n)$: A-priori probability for object o_n
  - $p(M) = p(m_1, m_2, \ldots, m_z)$: A-priori probability for measurement set M
  - $p(M|o_n)$: Probability density function for object o_n.
- Naive Bayes Assumption: $p(M|o_n)$ can not be calculated, but if all measurements m_k in M are assumed to be independent, we have:

$$p(M|o_n) = p(m_1|o_n) \cdot p(m_2|o_n) \cdots p(m_z|o_n) = \prod_k p(m_k|o_n). \tag{11}$$

  The factors $p(m_k|o_n)$ are estimated from the multidimensional histograms in the training phase.
- The a-priori probabilities $p(o_n)$ are assumed to be the same for all of the N objects o_n: $p(o_n) = 1/N$
- Then equation (10) reduces to

$$p(o_n|M) = \frac{\prod_k p(m_k|o_n)}{\sum_i^N \prod_k p(m_k|o_i)} \tag{12}$$

Naive Bayes Algorithm for Image Recognition

Naive Bayes Recognition based on multidimensional receptive fields

1. Generate multidimensional histograms $p(m_k|o_n)$ for all relevant objects.
2. Sample the query-image to obtain set of measurements M
3. For all objects o_n apply equation (12) in order to calculate a-posteriori probability $p(o_n|M)$ that given Measurements M correspond to object o_n.
4. Select object with highest a-posteriori probability, or reject query, if all a-posteriori probabilities are below a certain threshold.

Main Advantage of this probabilistic approach: As experiments have shown only small number z of local measurements $M = \{m1, \ldots, m_z\}$ are necessary for reliable recognition. I.e. the method is robust in the case of partial occluded objects.

Exercise: Probabilistic Recognition with 1D Histograms

In a new picture the 4 values 2, 5, 6, 2 have been measured. Which of the 3 objects (described by the above measurement histograms) is the most probable?

Experimental Results from [3]

- 103 different objects in 2130 images:
  - 83 objects in 690 images. Each image contains different scale or different rotation of object
  - 20 objects in 1440 images. 72 view points for each object (http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php)
- 6-dimensional histogram: D_x and D_y descriptor, each at 3 different scales $\sigma \in \{1, 2, 4\}$; Number of bins per histogram dimension: 24.

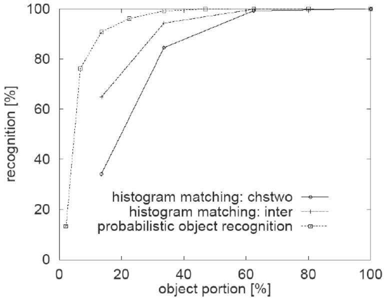
Experimental Results from [3]

Compare:

- Nearest Neighbor with *intersection* histogram distance measure.
- Nearest Neighbor with χ^2 histogram distance measure.
- Naive Bayes Recognition

Results:

- At 62% object visibility all 3 approaches have 100% correctness.
- Intersection matching more stable than χ^2 in the case of occluded objects.
- Probabilistic Recognition requires only 20% object visibility.



References I

[1] Y. Rubner, C. Tomasi, and L.J. Guibas. A metric for distributions with applications to image databases. In *ICCV*, pages 59–66, 1998.

[2] B. Schiele and J.L. Crowley. Object recognition using multidimensional receptive field histograms. In *ECCV (1)*, pages 610–619, 1996.

[3] B. Schiele and J.L. Crowley. Recognition without correspondence using multidimensional receptive field histograms. *International Journal of Computer Vision*, 36(1):31–50, 2000.

[4] M.J. Swain and D.H. Ballard. Color indexing. *International Journal of Computer Vision*, 7:11–32, 1991.