

Image Retrieval Foundations & Trends

Dr. Muhammad Sajjad

R.A: Kaleem Ullah

Overview

- Image contents
- Image Retrieval
- Motivation
- Content Based Image Retrieval
- Main Objective
- Features Extraction
- Feature Descriptors
 - Color Descriptors
 - Texture Descriptors
 - Shape Descriptors
 - Hybrid Descriptors
- Descriptor Matching Schemes

Image Contents

- A digital image is simply a collection of pixels arranged in the form of a matrix
- Individual pixels are merely numbers and carry no meaningful information
- Pixels combine to form different meaningful objects
- Groups of pixels that make up something meaningful in an image can be referred to as Contents.
- Contents are: colors, textures, object shapes, and any other information that can be derived from the image itself.

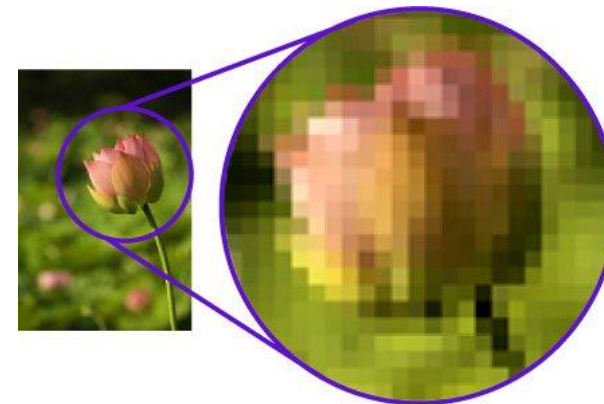
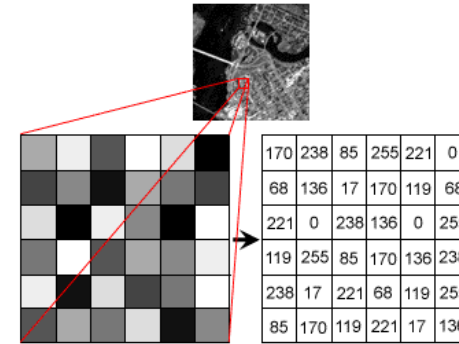


Image Retrieval

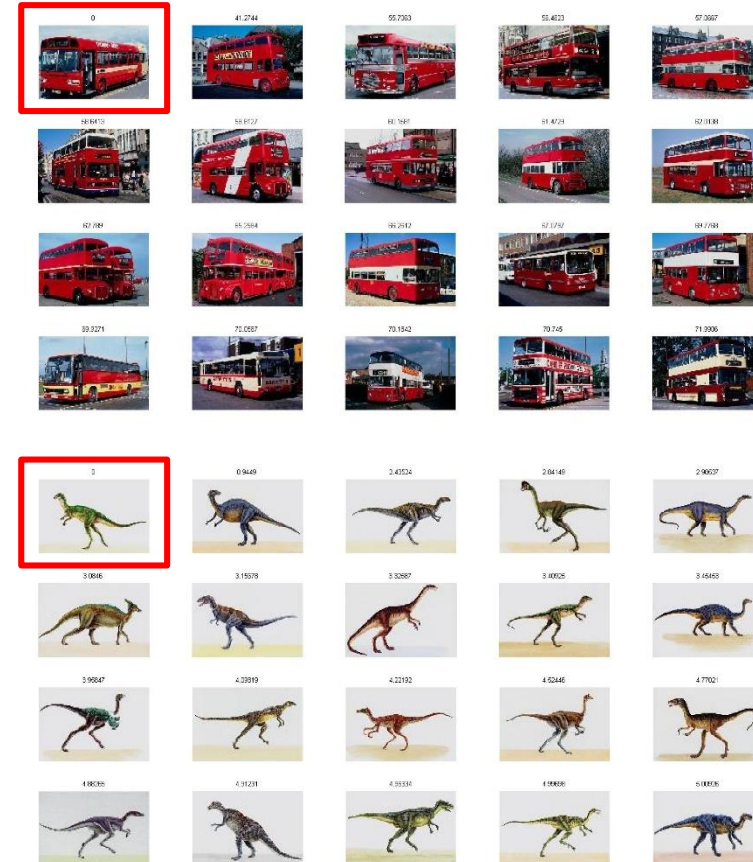
- The process of searching a desired set of images from a huge collection of different images is called image retrieval.
- Image retrieval can be achieved by two methods
 1. Text-Based Image Retrieval
 - Annotations, labels, and metadata are manually associated with images.
 - These annotations are used to locate desired images by performing text-based matching
 2. Content-Based Image Retrieval
 - No manual text annotations are needed
 - Images are retrieved based on their similarity derived from within their contents

Motivation

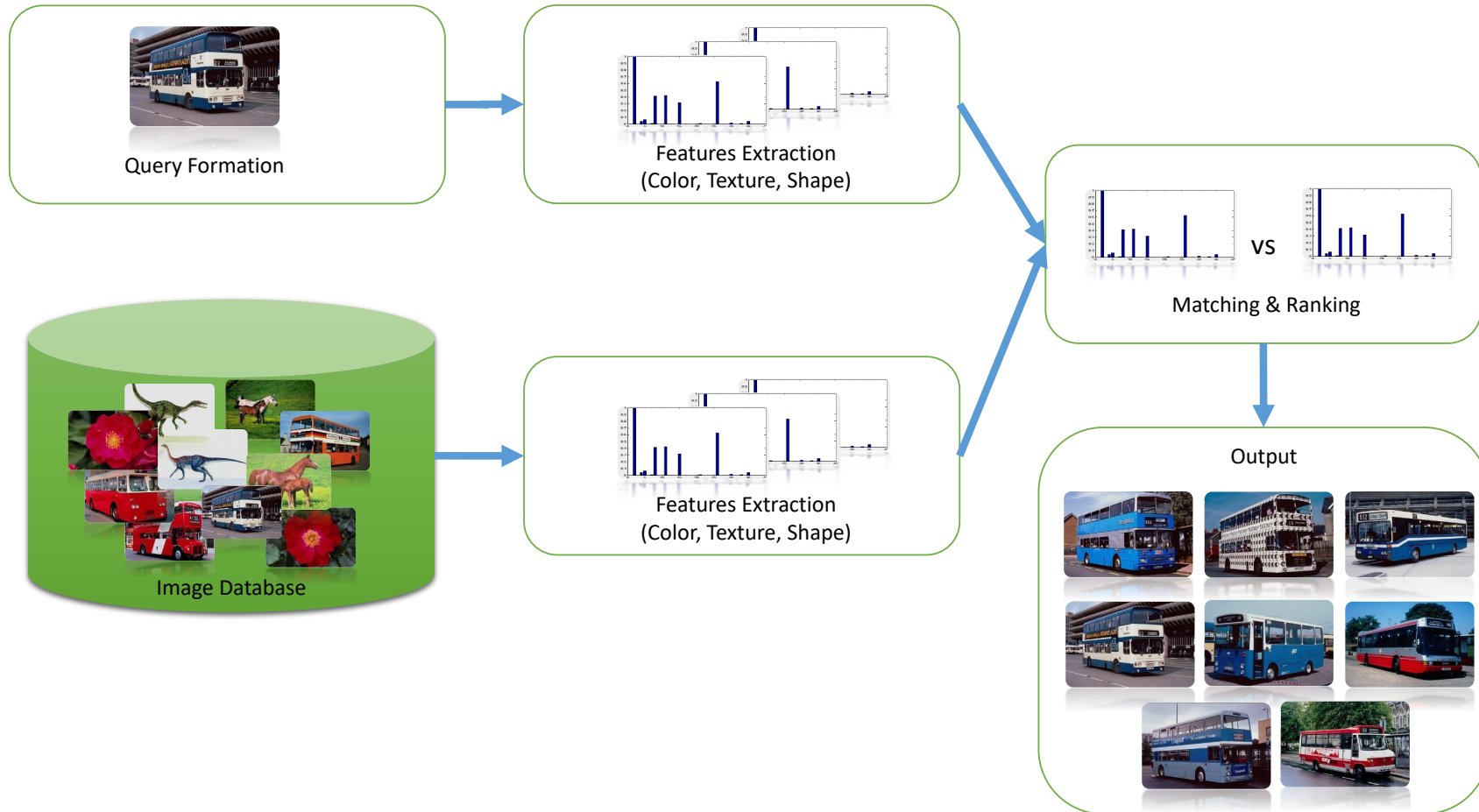
- Increase in computing power, inexpensive and easy availability of image capture devices, and cheap electronic storage capacity
- Increased use of image and video:
 - Entertainment
 - Education
 - Commercial purposes
- Exponential increase in digital image/video database sizes
- Need abstractions for efficient and effective browsing
- Manual annotations have become infeasible

Content Based Image Retrieval

- The retrieval of relevant images from an image database on the basis of automatically-derived image features
- A typical CBIR system consist of:
 - Feature extraction
 - Indexing
 - Retrieval
- Challenge:
 - Gap between low-level features and high level user semantics



Content Based Image Retrieval



Foundations

Main Objective

- The problem we are trying to solve is image similarity.
- Given two images (or image regions) – are they similar or not ?



- Solution: **Image Descriptors**
- An image descriptor “describes” a region in an image using the characteristics (features) of the image.
- To compare two such regions we will compare their descriptors.

Features Extraction

- In machine learning, pattern recognition and in image processing, feature extraction starts from an initial set of measured data and builds derived values (features) intended to be informative and non-redundant, providing better human interpretations.
- When the input data to an algorithm is too large to be processed and it is suspected to be redundant (e.g. images presented as pixels),
- then it can be transformed into a reduced set of features (also named a "features vector"). This process is called feature extraction.
- The extracted features are expected to contain the relevant information from the input data, so that the desired task can be performed by using this reduced representation instead of the complete initial data.

Feature Descriptors

- Primary Feature Descriptors
 - A. Color Descriptors
 - B. Texture Descriptors
 - C. Shape Descriptors
 - D. Hybrid Descriptors

A. Color Descriptors

1. Conventional color histogram (CCH)
 - Easy computation
 - Does not encode spatial info
 - Does not encode color pixel similarity
2. Color correlogram
 - Encode color layout as probabilities of occurrence of color pairs at a fixed distance
 - Easy computation
3. Color Structure Descriptor
 - Populates a histogram by taking local distributions of colors in the image
4. Dominant Color Descriptor
 - Encodes dominant colors into triads of **color values**, **percentages of colors** and their **variances**

1. Conventional Color Histogram

- A color histogram is a representation of the distribution of colors in an image
- It shows different types of colors appeared and the number of pixels in each type of the colors appeared
- Color histogram focuses only on the proportion of the number of different types of colors, regardless of the spatial location of the colors
- Number of colors are usually reduced for easy computation
- For example: A color histogram in the RGB color space can be represented using four bins. Bin 0 corresponds to intensities 0-63, bin 1 is 64-127, bin 2 is 128-191, and bin 3 is 192-255.

Red	Green	Blue	Pixel Count
0	0	0	7414
0	0	1	230
0	0	2	0
0	0	3	0
0	1	0	8
0	1	1	372
0	1	2	88
0	1	3	0
0	2	0	0
0	2	1	0
0	2	2	10
...
3	2	3	109
3	3	0	0
3	3	1	0
3	3	2	3415
3	3	3	53929

2. Color Correlogram

- To overcome the deficiencies in color histograms
- It includes the spatial correlation of colors
- It can be used to describe the global distribution of local spatial correlation of colors
- It is easy to compute
- The size of the feature is fairly small
- The auto-correlogram of image I for color C_i , distance k :

$$\gamma_{C_i}^{(k)}(I) \equiv \Pr[p_2 \in I_{C_i} \mid p_1 \in I_{C_i}, |p_1 - p_2| = k]$$

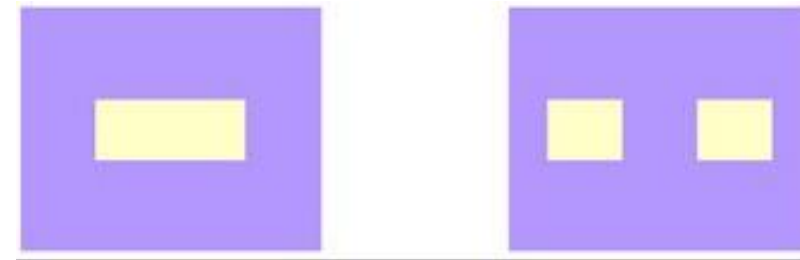
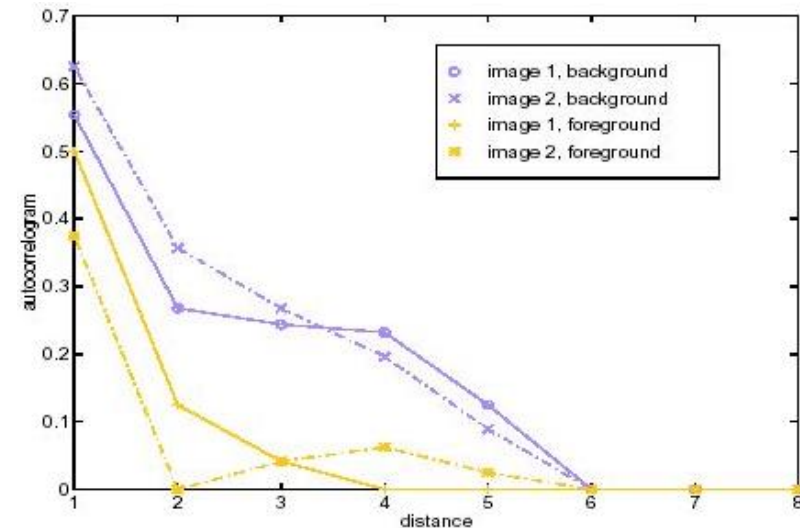


Image 1

Image 2



3. Color Structure Descriptor

- It expresses local color structure in an image by use of a structuring element
- The CSD is computed by visiting all location in the image, retrieving colors $C\{0-7\}$ of all pixels contained in the 8x8 pixel structure element.
- CSD bine of each color contained inside the window are incremented.

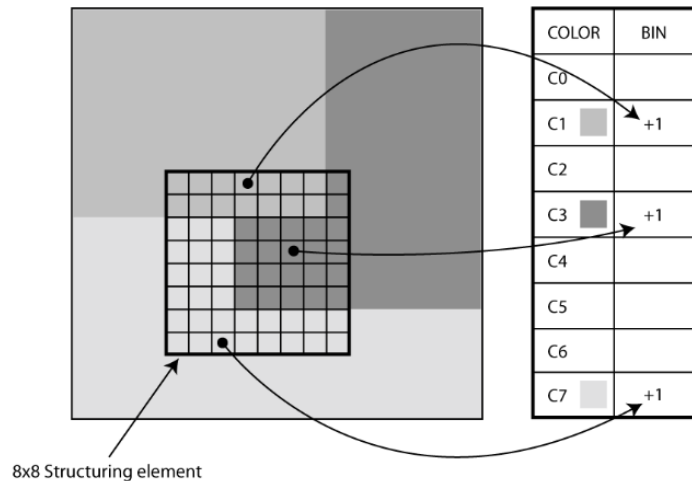


Figure 1. – CSD structuring element

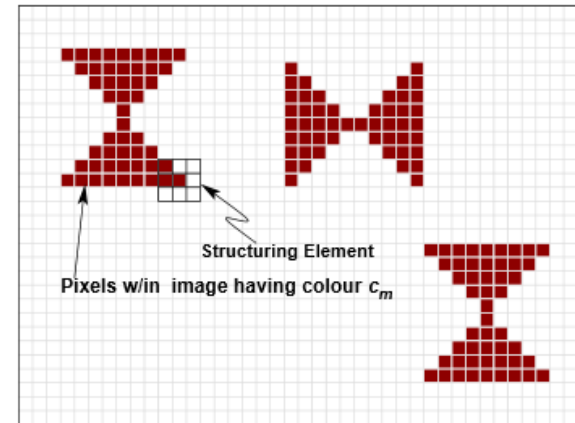


Fig. 1. Highly structured (coherent) iso-colour plane consisting of 150 pixels with colour c_m . For $s = 9$ we have $h_s(m) = 324$.

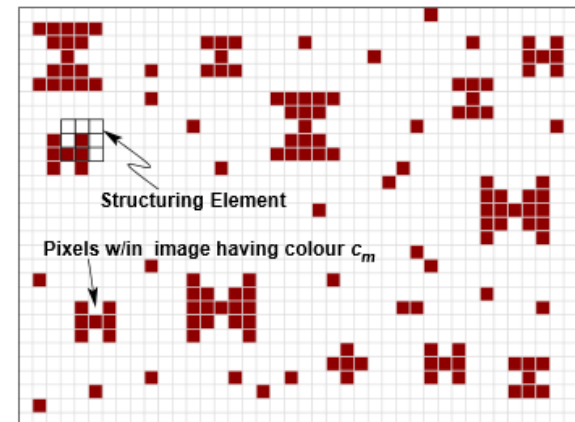





Fig. 2. Un-structured (incoherent) iso-colour plane c_m of 150 pixels. For $s = 9$ we have $h_s(m) = 589$.

4. Dominant Color Descriptor

- The dominant color descriptor is defined as:
- $F = \{C_i, P_i\}, (i=1,2,3...N)$
- Where C_i is a 3D dominant color vector, P_i is the percentage of each dominant color.
- $C = 9 \cdot H + 3 \cdot S + V; C \in [0, 71]$
- Calculate the quantified HSV space histogram, with $P_i (i=0,1,..,71)$ express proportion of color vector i ;
- Similarity measure:

$$D(F_Q, F_I) = \sum_{i=0}^{71} \min(P_{Qi}, P_{Ii});$$

Query image Q	Target image F1	Target image F2
$\{(33,31,33), 0.794240\}$	$\{(66,41,29), 0.108795\}$	$\{(60,55,53), 0.378306\}$
$\{(184,179,180), 0.20576\}$	$\{(203,47,71), 0.334035\}$	$\{(139,123,115), 0.073598\}$
	$\{(207,193,59), 0.067861\}$	$\{(198,194,188), 0.548096\}$
	$\{(228,98,161), 0.219045\}$	
	$\{(230,162,203), 0.270264\}$	
		
		

B. Texture Descriptors

1. Statistical Texture Features

- Smoothness
- Coarseness
- Regularity

2. Edge Histogram Descriptor

- Represent textures by local edge characteristics

3. Texture Browsing Descriptor

- captures texture characteristics like directionality, coarseness and regularity

4. Local Binary Patterns Histogram

- Analyzes textures in small neighborhoods
- Computes binary values from the pixel arrangements in neighborhood

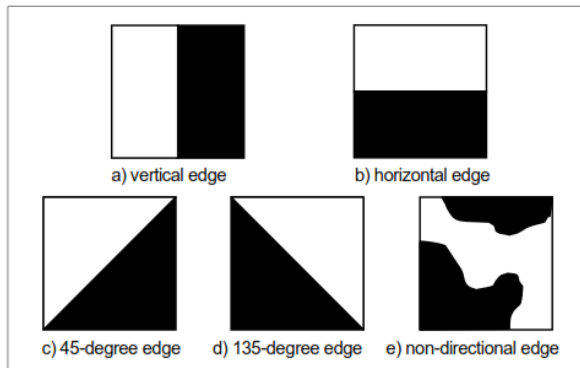
1. Statistical Texture Features

Moment	Expression	Measure of Texture
Mean	$m = \sum_{i=0}^{L-1} z_i p(z_i)$	A measure of average intensity.
Standard deviation	$\sigma = \sqrt{\mu_2(z)} = \sqrt{\sigma^2}$	A measure of average contrast.
Smoothness	$R = 1 - 1/(1 + \sigma^2)$	Measures the relative smoothness of the intensity in a region. R is 0 for a region of constant intensity and approaches 1 for regions with large excursions in the values of its intensity levels. In practice, the variance used in this measure is normalized to the range $[0, 1]$ by dividing it by $(L - 1)^2$.
Third moment	$\mu_3 = \sum_{i=0}^{L-1} (z_i - m)^3 p(z_i)$	Measures the skewness of a histogram. This measure is 0 for symmetric histograms, positive by histograms skewed to the right (about the mean) and negative for histograms skewed to the left. Values of this measure are brought into a range of values comparable to the other five measures by dividing μ_3 by $(L - 1)^2$ also, which is the same divisor we used to normalize the variance.
Uniformity	$U = \sum_{i=0}^{L-1} p^2(z_i)$	Measures uniformity. This measure is maximum when all gray levels are equal (maximally uniform) and decreases from there.
Entropy	$e = - \sum_{i=0}^{L-1} p(z_i) \log_2 p(z_i)$	A measure of randomness.

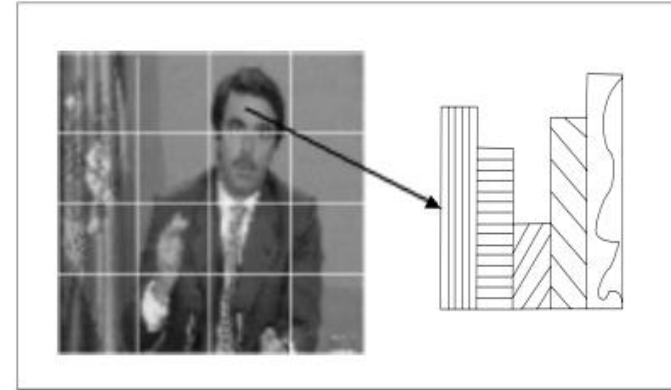
Texture	Average Intensity	Average Contrast	R	Third Moment	Uniformity	Entropy
Smooth	87.02	11.17	0.002	-0.011	0.028	5.367
Coarse	119.93	73.89	0.078	2.074	0.005	7.842
Periodic	98.48	33.50	0.017	0.557	0.014	6.517

2. Edge Histogram Descriptor

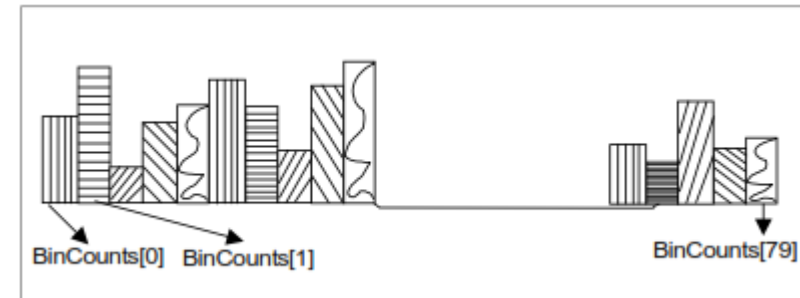
- Edges in images constitute an important feature to represent their content.
- Human eyes are sensitive to edge features for image perception.



Five type of edges



Divide the image into 4x4 regions. Compute edge pixels for each edge type and populate a histogram



For 16 subimages, 80 bin histogram is obtained. This histogram is referred to as Edge Histogram Descriptor or Edge Orientation Histogram.

3. Texture Browsing Descriptor

- This is a compact descriptor that requires only 12 bits (maximum) to characterize a texture's regularity (2 bits), directionality (3 bits x 2), and coarseness (2 bits x 2).
- A texture may have more than one dominant direction and associated scale.

Regularity: 11, 10, 01, 00

Directionality: quantized into six values ranging from 0° to 150° in steps of size 30° .

Coarseness: quantized into 4 values.

Coarsest = 3, fine-grained = 0

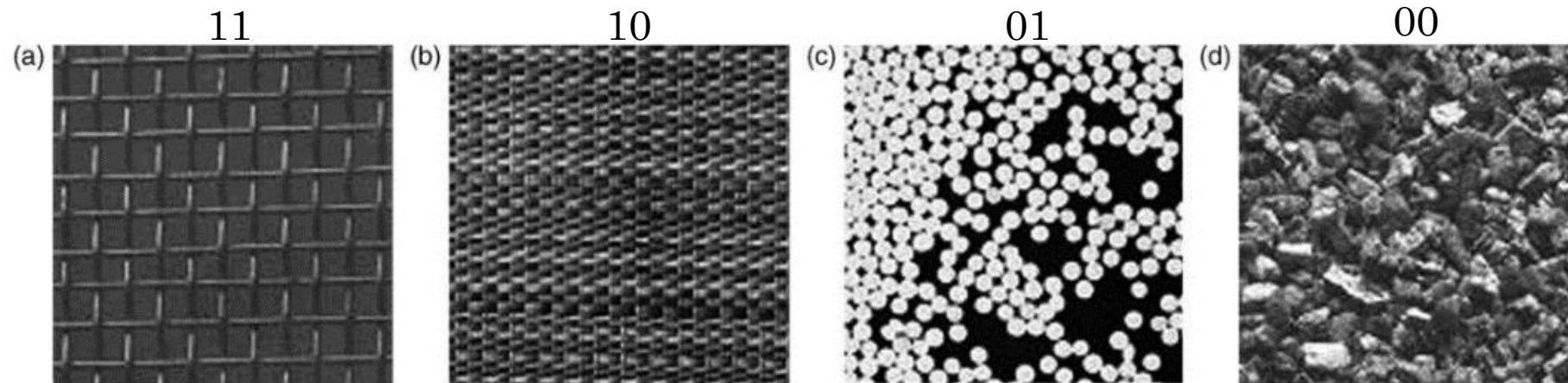
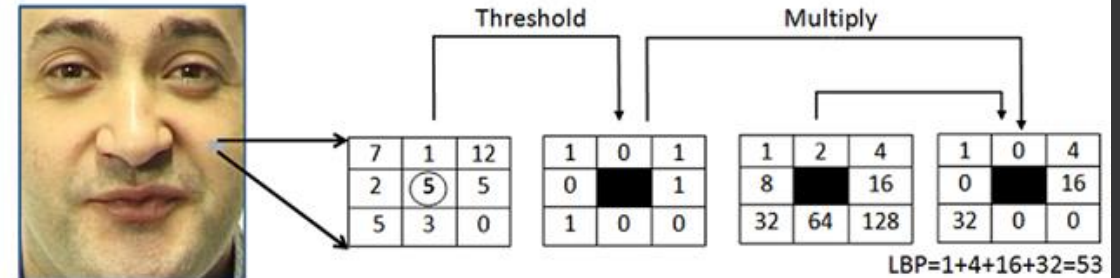


Fig. 2. An example of regularity classification. (a) Highly regular. (b) Regular. (c) Slightly regular. (d) Irregular.

4. Local Binary Patterns

- An efficient texture operator
- Labels pixels of an image by thresholding the neighborhood of each pixel and
- Considers the result as a binary number
- Histogram is populated from the computed values
- This histogram characterizes the texture and hence can be used for texture classification.



C. Shape Descriptors

1. Shape Signature

- One-dimensional functions derived from the shape's contour

2. Fourier Descriptors

- Fourier shape descriptors are calculated by applying Fourier Transform on 1D shape signatures discussed previously which are usually derived from the shape's contour

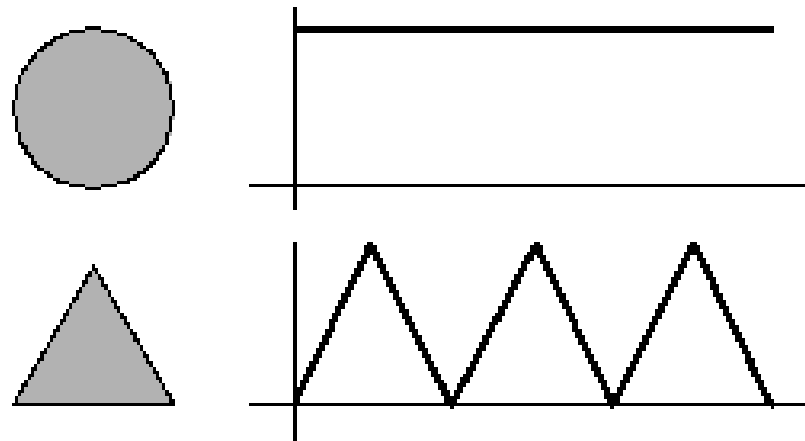
3. Invariant Moments

- Moments are basically the quantitative measure of the shape of a set of points

1. Shape Signature

- Signature is a 1D functional representation of a boundary and may be generated in various ways.
- One of the simplest is to plot the distance from an internal point (e.g. the centroid) to the boundary as a function of angle.

$$r(n) = [x(n) - gx]^2 + (y(n) - gy)^2]^{\frac{1}{2}}$$



2. Fourier Descriptors

- The Fourier shape descriptors are calculated by applying Fourier Transform on 1D shape signatures discussed previously which are usually derived from the shape's contour.

$$a_n = \frac{1}{N} \sum_{u=0}^{N-1} z(u) \cdot e^{-j2\pi nu/N}, \quad n=0, 1, \dots, N-1$$

- The Fourier coefficients obtained a_n (also called FDs) are normalized by using the DC component of the transform as

$$f = \left[\frac{|FD_2|}{|FD_1|}, \frac{|FD_3|}{|FD_1|}, \dots, \frac{|FD_{N-1}|}{|FD_1|} \right]$$

- Where FD1 is the DC component, and FD_N are the Fourier coefficients.

3. Invariant Moments

- Moments are basically the quantitative measure of the shape of a set of points. These moments are also known as Invariant Moments.
- Geometric moment function m_{pq} of order $(p+q)$ are given by:

$$m_{pq} = \sum_x \sum_y x^p y^q f(x, y) \quad p, q = 0, 1, 2 \dots$$

- The translation invariant geometric central moments are defined as:

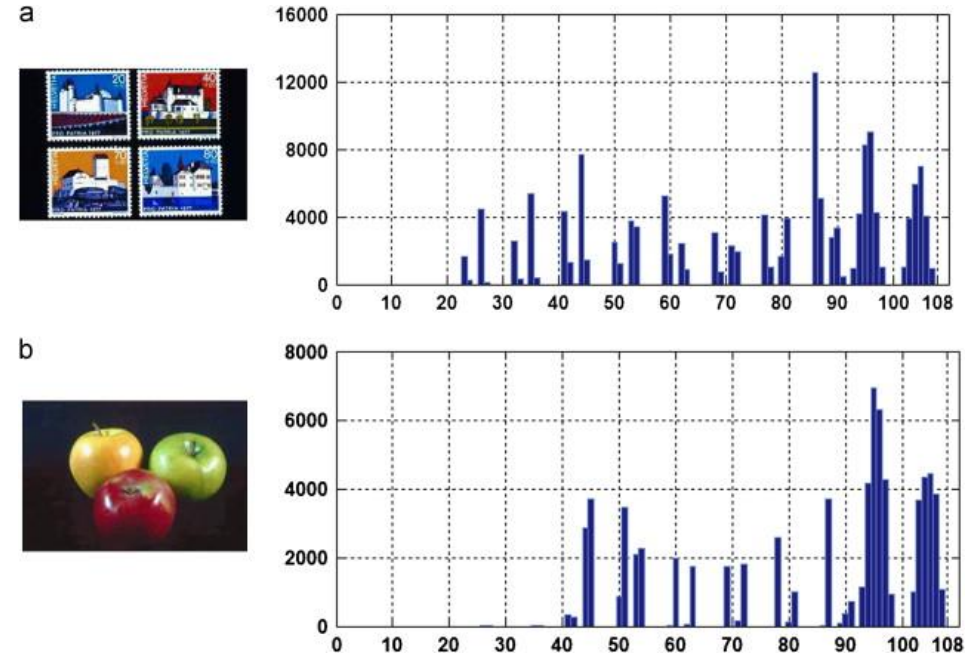
$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad p, q = 0, 1, 2 \dots$$

D. Hybrid Descriptors

1. Color Difference Histogram
2. Fuzzy Color and Texture Histogram
3. Color and Edge Directivity Descriptor
4. Micro Structure Descriptor

1. Color Difference Histogram

- Exploits color and edge orientations along with perceptually uniform color differences to encode these features into a manner that is similar to the human vision system.
- Images are quantized in $L^*a^*b^*$ color space into 90 colors ($L = 10$ bins, $a = 3$ bins, $b = 3$ bins)
- Perceptually uniform color difference between neighboring color indexes with edge orientation information as a constraint leads to an 18-dimensional vector;
- In total, $90+18=108$ -dimensional vector is obtained for the final image features during image retrieval.



2. Fuzzy Color and Texture Histogram

- Fuzzy system was proposed in order to produce a fuzzy-linking histogram, which regards the three channels of HSV as inputs, and forms a 10 bins histogram as an output. Each bin represents a preset color as follows: (0) Black, (1) Gray, (2) White, (3) Red, (4) Orange, (5) Yellow, (6) Green, (7) Cyan, (8) Blue and (9) Magenta.
- Texture features are represented as high frequency bands of wavelet transforms (histogram of 8 bins)
- (0) Low Energy Linear area, (1) Low Energy Horizontal activation, (2) Low Energy Vertical activation, (3) Low Energy Horizontal and Vertical activation, (4) High Energy Linear area, (5) High Energy Horizontal activation, (6) High Energy Vertical activation, (7) High Energy Horizontal and Vertical activation.

3. Color and Edge Directivity Descriptor

- Fuzzy system was proposed in order to produce a fuzzy-linking histogram, which regards the three channels of HSV as inputs, and forms a 10 bins histogram as an output. Each bin represents a preset color as follows: (0) Black, (1) Gray, (2) White, (3) Red, (4) Orange, (5) Yellow, (6) Green, (7) Cyan, (8) Blue and (9) Magenta.
- Five digital filters that were proposed by the MPEG-7 Edge Histogram Descriptor - EHD
- These filters are used for the extraction of the texture's information

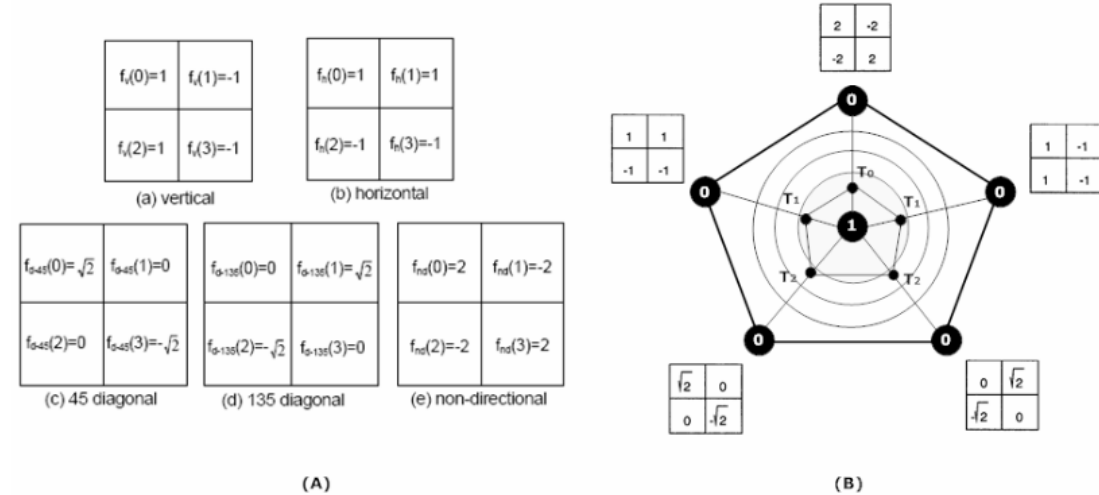
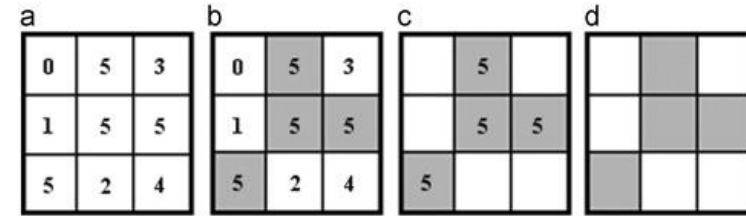


Fig. 3. (A) Filter coefficients for edge detection [7], (B) Edge Type Diagram

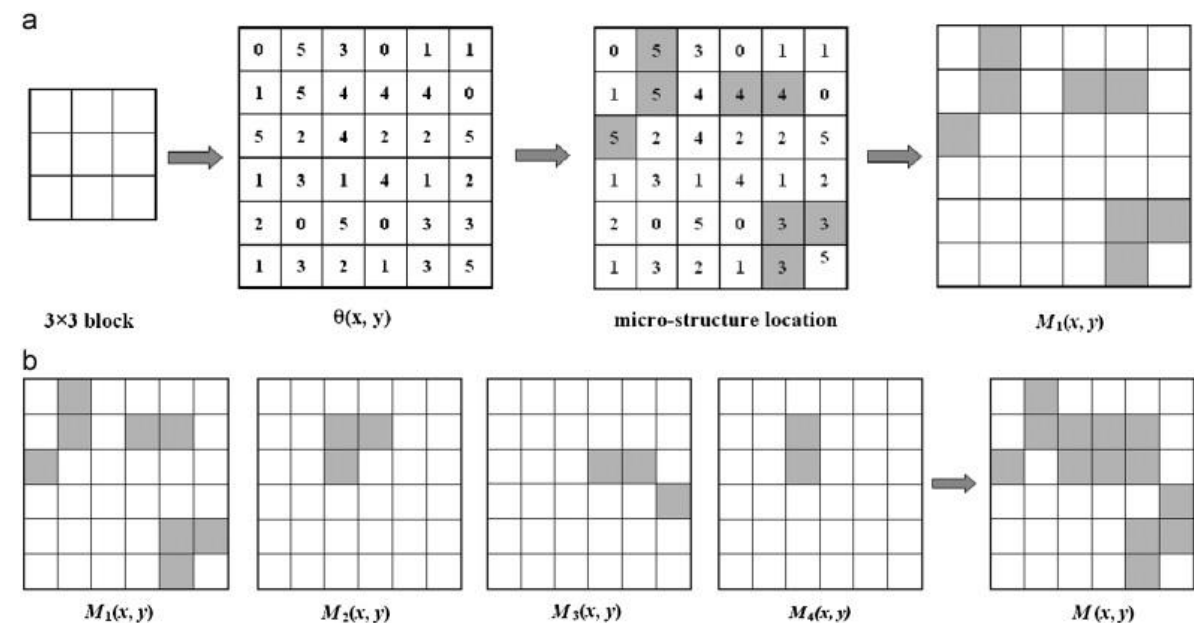
4. Micro Structure Descriptor

- Structural approach assumes that texture is formed with simple primitives called “texels” (texture elements) by following some placement rules
- Extracts micro-structures and describes them effectively in a 72 bin histogram
- Moving the 3×3 block from left-to-right and top-to-bottom throughout the micro-structure image, we use the following equation to describe the micro-structure features

$$H(w_0) = \begin{cases} \frac{N\{f(p_0)=w_0 \wedge f(p_i)=w_i | |p_i-p_0|=1\}}{8N\{f(p_0)=w_0\}} \\ \text{where } w_0 = w_i, i \in \{1, 2, \dots, 8\} \end{cases}$$



An example of micro-structure detection



Descriptor Matching

- Visual contents represented as descriptors are used to derive similarity between images.
- Several metrics exist for computing the similarity or difference between descriptors.
- Common metrics include:

1. Euclidean Distance: $d(\mathbf{p}, \mathbf{q}) = d(\mathbf{q}, \mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2}$

$$= \sqrt{\sum_{i=1}^n (q_i - p_i)^2}.$$

2. Manhattan Distance: $d_1(\mathbf{p}, \mathbf{q}) = \|\mathbf{p} - \mathbf{q}\|_1 = \sum_{i=1}^n |p_i - q_i|,$

3. Mahalanobis Distance: $d(\underline{x}, \underline{y}) = \sqrt{\sum_{i=1}^N \frac{(x_i - y_i)^2}{s_i}},$ where s is variance of x and y

Future Trends

Semantic Gap

- Visual similarity does not always mean semantic similarity
- Semantically similar may not be visually similar



(a)

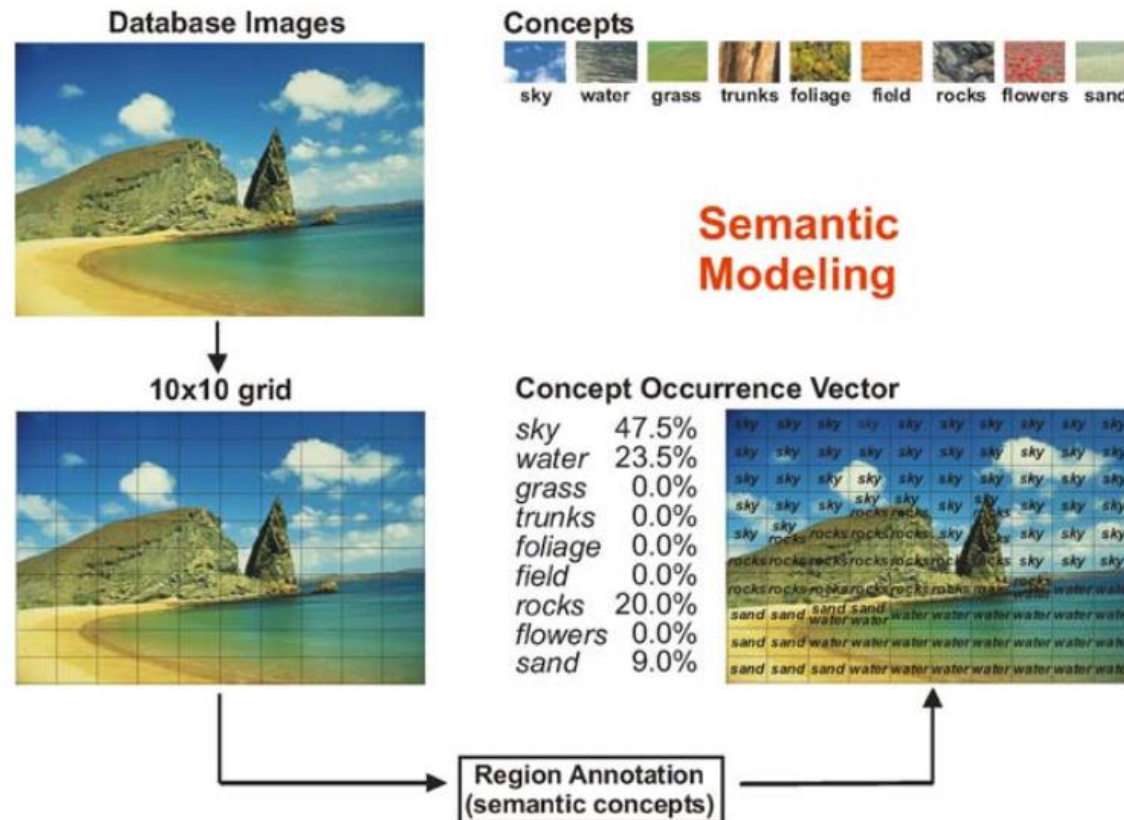


(b)

An illustration of “semantic gap”

Semantic Image Representation

- High level representation of images
- Associate high level concepts with images automatically



Feature Learning

- Feature learning or representation learning is a set of techniques that learn a feature: a transformation of raw data input to a representation that can be effectively exploited in machine learning tasks.
- Feature learning can be divided into two categories: supervised and unsupervised feature learning, analogous to these categories in machine learning generally.
 - In supervised feature learning, features are learned with labeled input data. Examples include neural networks, multilayer perceptron, and (supervised) dictionary learning.
 - In unsupervised feature learning, features are learned with unlabeled input data. Examples include dictionary learning, independent component analysis, auto-encoders, matrix factorization, and various forms of clustering.

Thank You