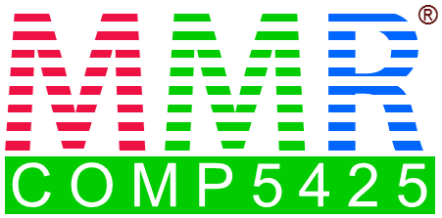


MULTIMEDIA RETRIEVAL

THE UNIVERSITY OF SYDNEY

Week05

Semester 1, 2025

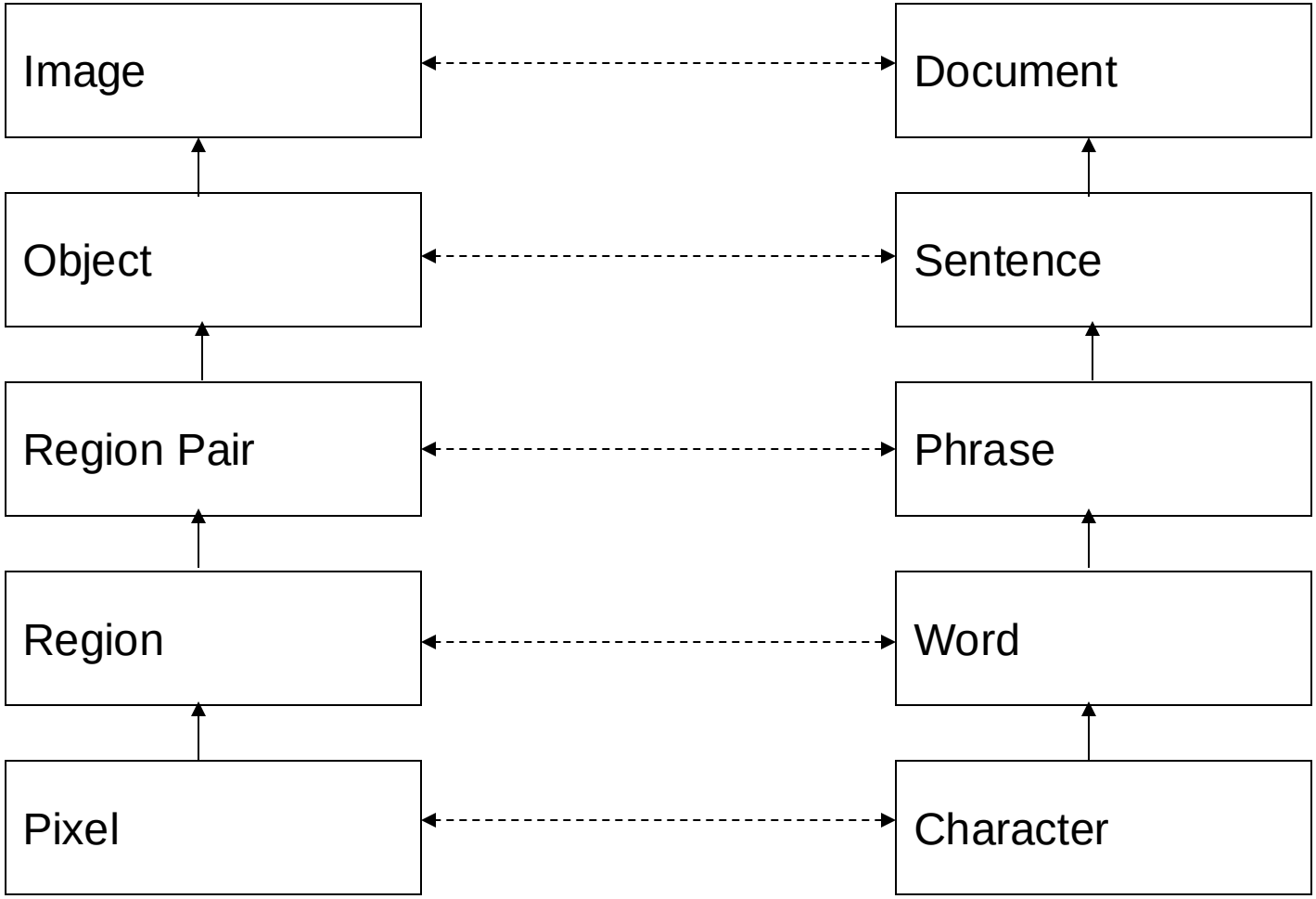
Content Based Retrieval I

- Background
- Visual feature extraction
  - Color
  - Texture
  - Shape

THE LORD OF THE RINGS
THE FELLOWSHIP OF THE RING

http://www.imdb.com/

Text vs. Image

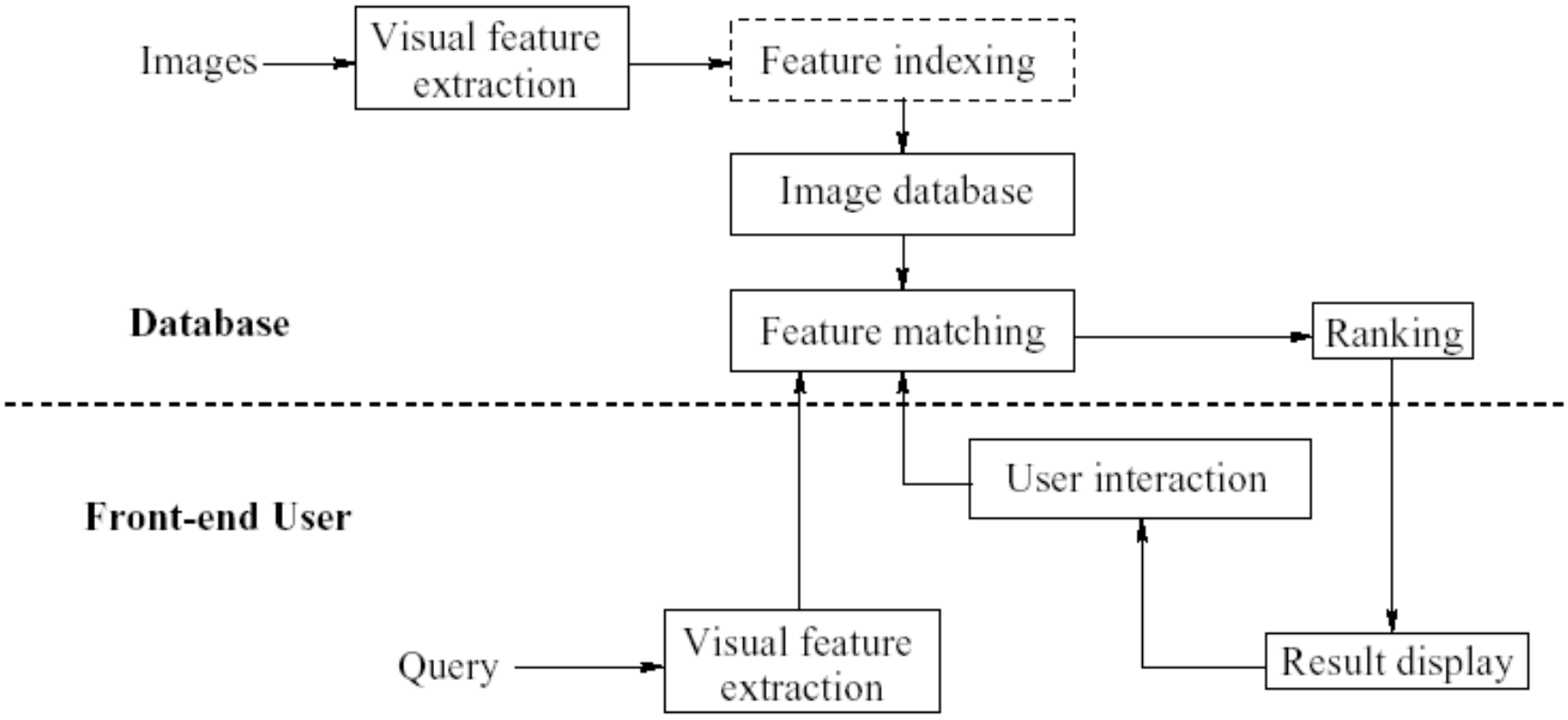


Content-based Retrieval

- Using visual features to reflect rich contents
  - Color, shape, texture, …
- Automatic feature extraction
- Plenty of applications
  - Query by example, query by sketch, …
- No explicit semantic concepts
- Difficult for indexing high dimensional features

A picture is worth a thousand words.

Content-based Retrieval

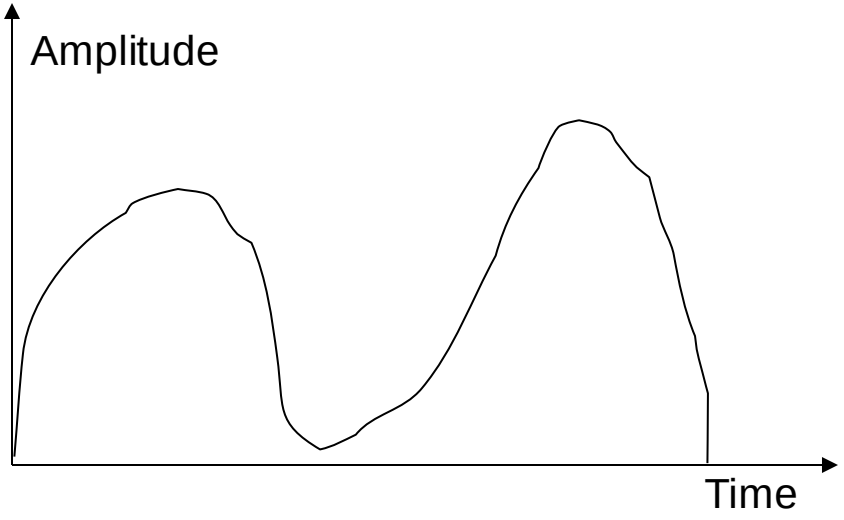


Content-based image retrieval diagram

Content-based Retrieval

- Feature extraction is performed to obtain multi-dimensional feature vectors characterizing multimedia contents.
  - Different media has different information representations.
- Appropriate similarity measurement is employed to measure the similarity between query item and database item.
- Feedback provides the interactions between users and systems.
- Efficient indexing techniques are employed to organize databases.
- Benchmarking is to evaluate retrieval performance.
  - TREC (Text REtrieval Conference) ???

Audio Representation



- Acoustic features
- Subjective / Semantic features

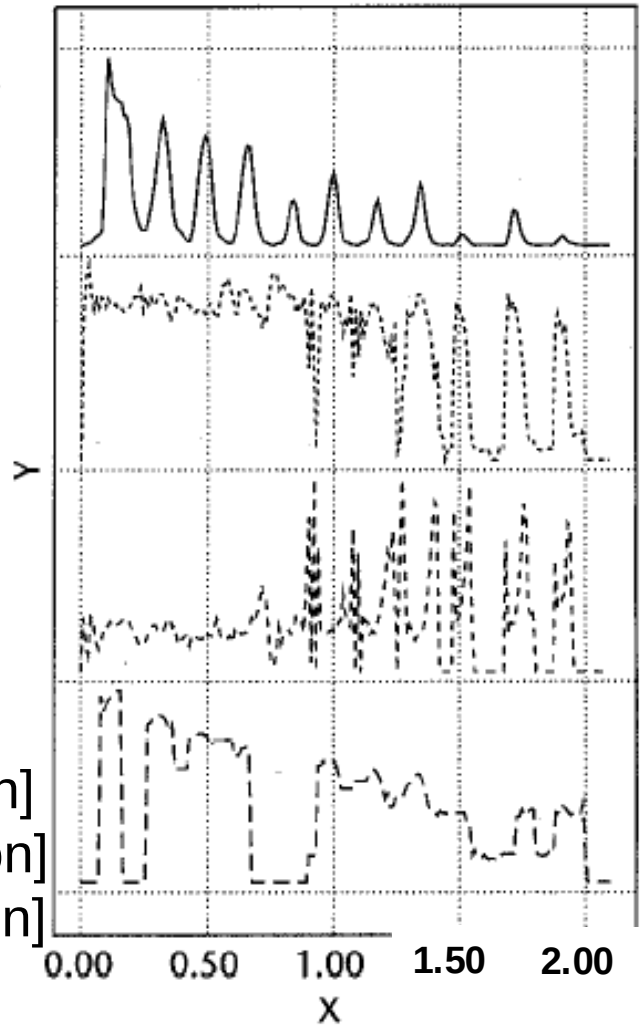
Acoustic features

- Loudness
  - The signal's root-mean-square (RMS) level in decibels over a series of windowed frames.
- Pitch
  - Pitch is estimated by taking a series of short-time Fourier spectra.
- Brightness
  - The centroid of the short-time Fourier magnitude spectra.
- Bandwidth
  - The magnitude-weighted average of the differences between the spectral components and the centroid.
- Harmony
  - It is computed by measuring the deviation of the sound's line spectrum from a perfectly harmonic spectrum. It distinguishes between harmonic spectra, inharmonic spectra, and noise.
- Energy

Refer to http://mpeg7.doc.gold.ac.uk/ for more MPEG-7 Audio Features

Acoustic features

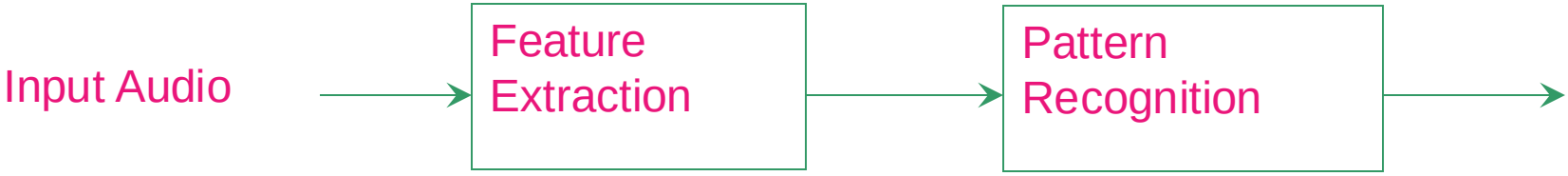
Male Laughter

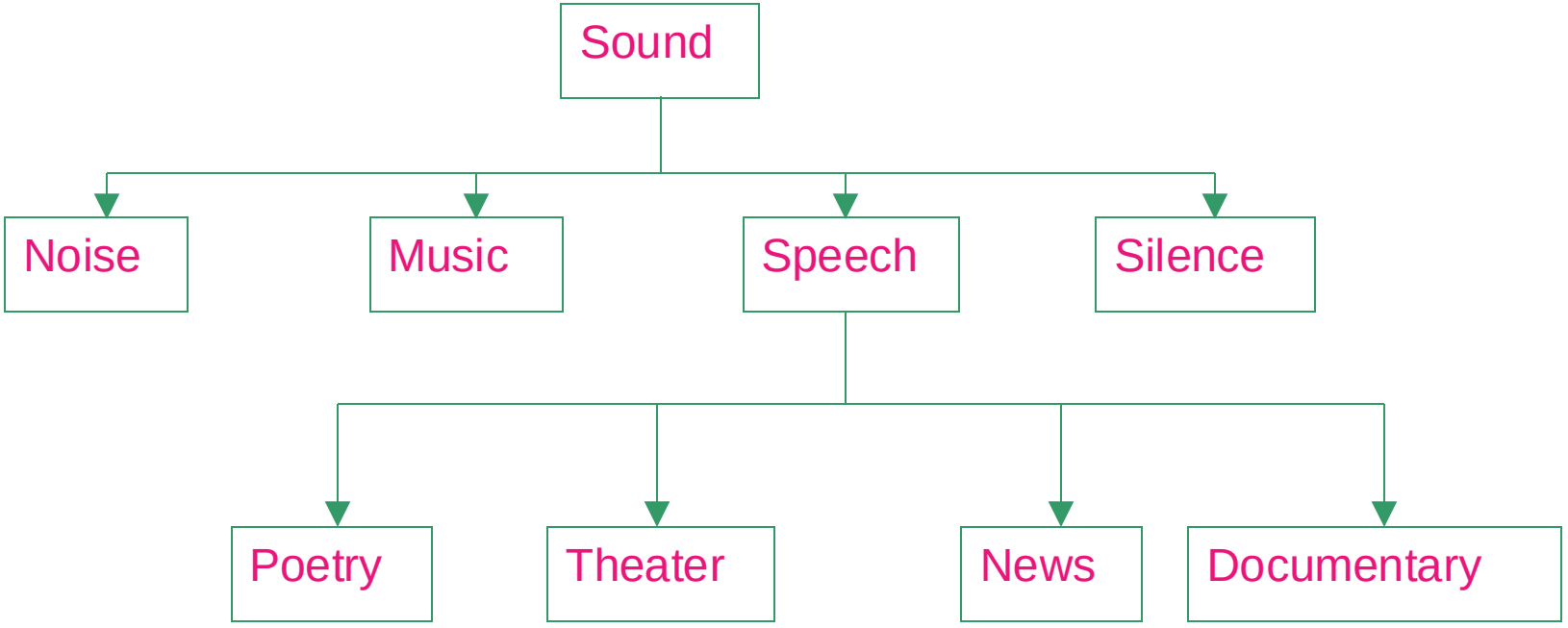


Feature Vectors:

1. Pitch [mean, variance, autocorrelation]
2. Amplitude [mean, variance, autocorrelation]
3. Brightness [mean, variance, autocorrelation]
4. Bandwidth [mean, variance, autocorrelation]

Audio Classification and Retrieval





Example: Shazam

Shazam is an application that can identify music based on a short sample played using the microphone on the device.

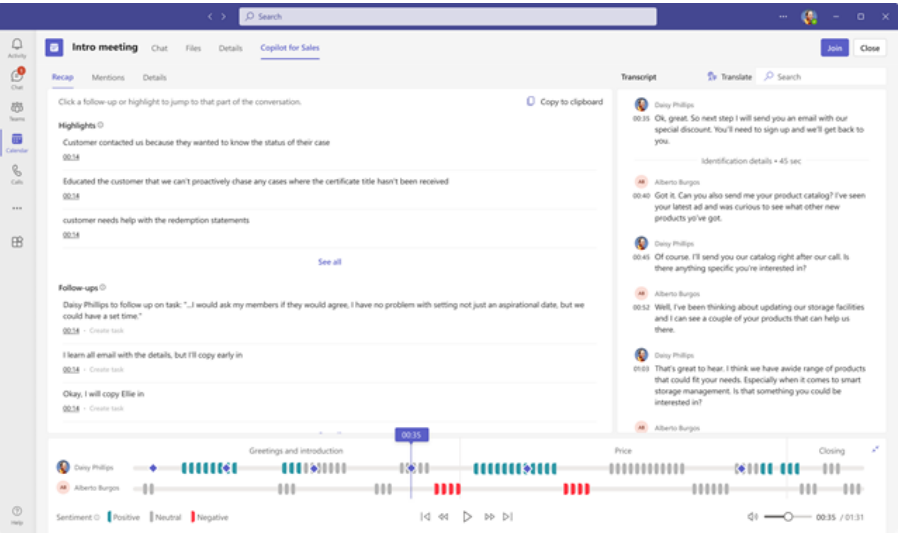
https://www.shazam.com/

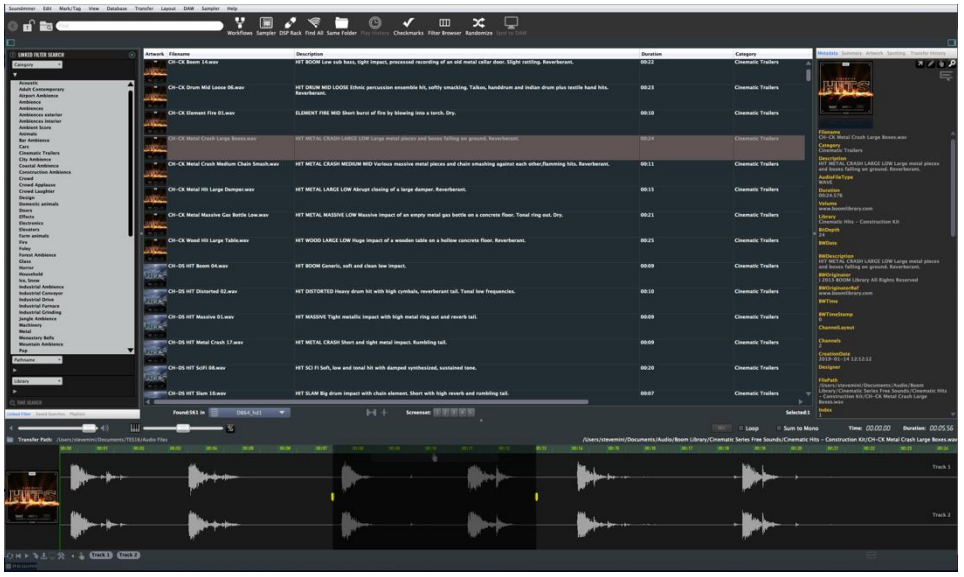
Applications of Audio Classification and Retrieval

- Audio database management
- Audio database browser
- Audio editor
- Assistance in video analysis
  - Surveillance, such as silence detection
  - Sports
  - Movie genre classification
- Speech recognition
- Music genre classification
- Instrument identification

Textbook pp.96-106

Applications





https://learn.microsoft.com/en-us/microsoft-sales-copilot/view-understand-meeting-summary

https://store.soundminer.com/

Image Representation

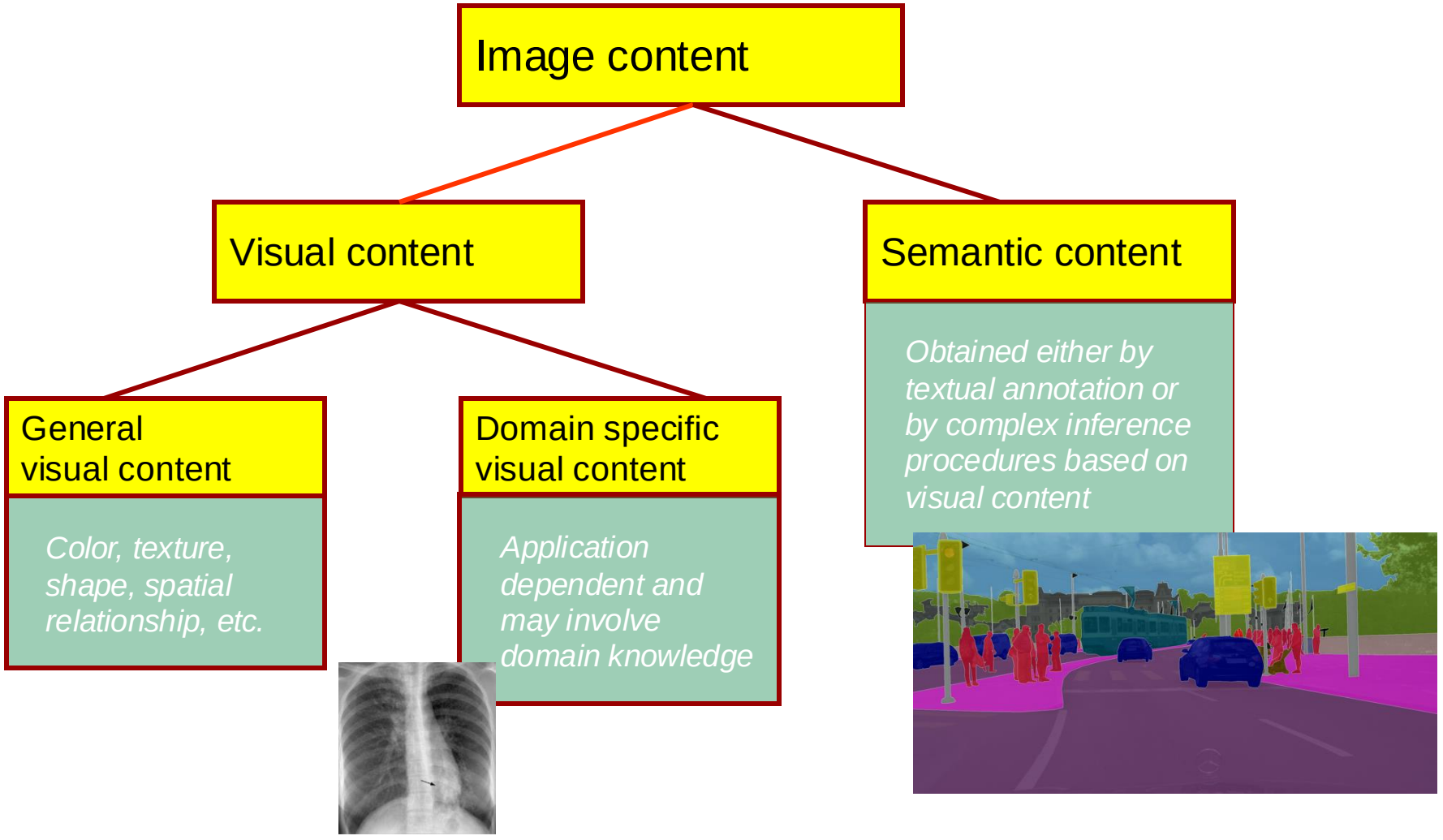


Image Representation

- A visual content descriptor can be either global or local.
- A global descriptor uses the visual features of the whole image.
- A local descriptor uses the visual features of regions or objects to describe the image content, with the aid of region segmentation and object segmentation techniques.

Image Representation

- Color
- Shape
- Texture
- Spatial relationship
- Others
  - Compression: fractal coding, JPEG

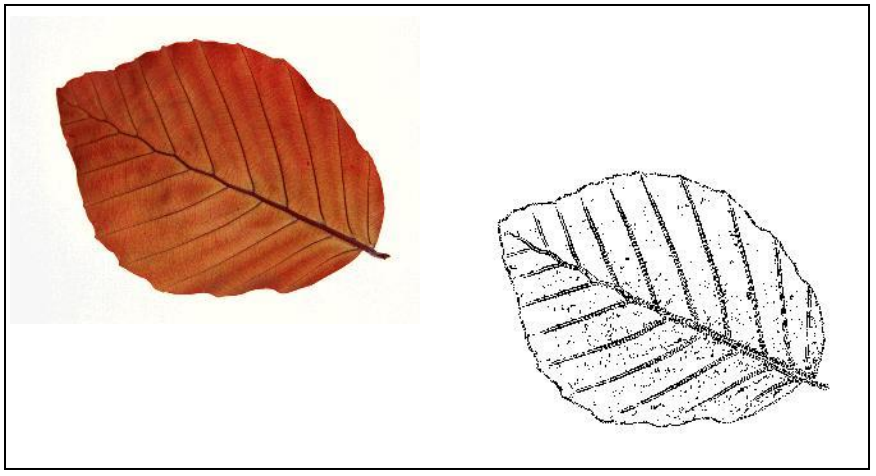



Figure No. 1
File Edit Window Help
Original image
p013_0_94.jpg
Similarity percentage
100
Figure No. 2
File Edit Window Help
1. match
p017_0_80.jpg
Similarity percentage
73
Figure No. 3
File Edit Window Help
2. match
p17_0_79.jpg
Similarity percentage
72.5
Figure No. 4
File Edit Window Help
3. match
p72_0_79.jpg
Similarity percentage
72.5
Figure No. 5
File Edit Window
Help
Figure N...
File Edit
Window Help
Figure No. 7
File Edit Window
Help
6. match
p094_0_98.jpg
Similarity percentage
66.5
Figure No. 8
File Edit Window
Help
7. match
p40_0_77.jpg
Similarity percentage
66
Figure No. 9
File Edit Window
Help
8. match
p55_0_85.jpg
Similarity percentag
65.5
Figure No....
File Edit Window
Help
9. match
p083_0_94.jpg
Similarity percentag
64.5
Figure N...
File Edit
Window Help
10. match
p041_0_80.jpg
Similarity percentage
64

Color

- Human is more sensitive about color.
- Color is very powerful in description and of easy extraction.
- Color varies considerably with the change of illumination, orientation of the surface, and the viewing geometry of the camera.

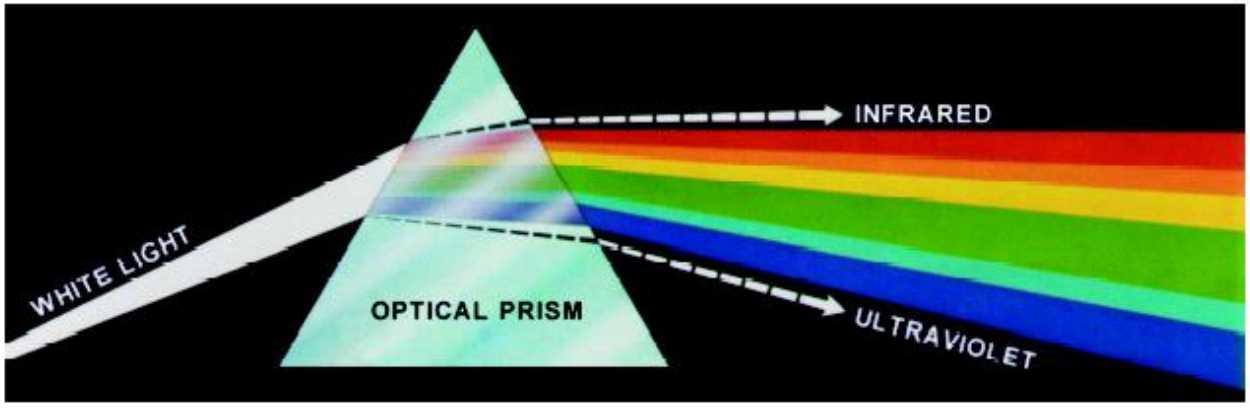
Color

- Color fundamentals
- Color spaces
- Color features
  - Color histogram
  - Color moments
  - Color coherence vector (CCV)
- Similarity between colors

Color Fundamentals

Light and the Electromagnetic (EM) Spectrum

- In 1666, Sir Isaac Newton discovered that when a beam of sunlight passes through a glass prism, the emerging beam of light is not white but consists instead of a continuous spectrum of colors ranging from violet at one end to red at the other.
- The color spectrum may be divided into six broad regions: violet, blue, green, yellow, orange, and red.



R. Gonzalez, R. Woods, “Digital Image Processing”, Prentice Hall, 2002.

Color Fundamentals

The human eye contains two different sorts of receptor cells: *rods*, which provide night-vision and cannot distinguish color, and *cones*, which are highly sensitive to color and in turn come in three different sorts, which respond to different wavelengths of light.

The fact that our perception of color derives from the eye's response to three different groups of wavelengths leads to the *tristimulus theory* – that any color can be specified by just three values, giving the weights of each of three components. We call **red** **green** and **blue** the *additive primary colors*

Color Space

- A color space (also called color model or color system) is a specification of a ***coordinate system*** and a subspace within that system where ***each color*** is represented by a ***single point***.

- Most color spaces in use today are oriented either toward hardware (e.g. for color monitors and printers) or toward applications where color manipulation is a goal (e.g. in the creation of color graphics for animation).

- Color space is used to represent and reproduce colors

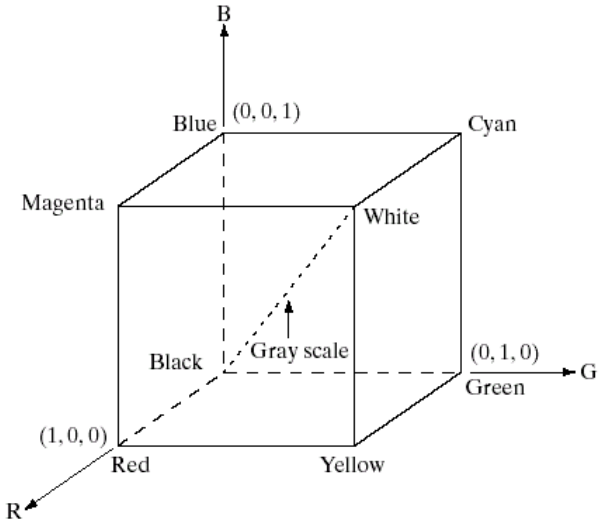
Color Space

- RGB (red, green, blue) space
  - The RGB color space is the most important means of representing colors used in images for multimedia, because it corresponds to the way in which color is produced on computer color monitors, and it is also how color is detected by scanners.
- HSV (hue, saturation, value) space
  - It corresponds closely with the way humans describe and interpret color
- CIE $L^*a^*b^*$ / CIE $L^*u^*v^*$ space
- CMYK (cyan, magenta, yellow, black) space
  - Better for color printing

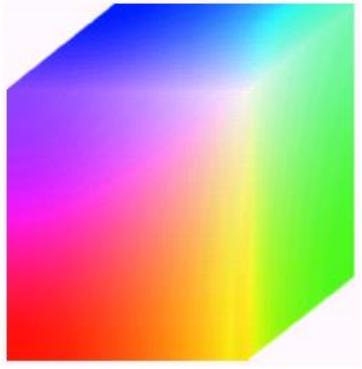
RGB

The RGB Color Space

- Each color appears in its primary spectral components of red, green, and blue.
- This space is based on a Cartesian Coordinate System.
- The color subspace of interest is the cube, in which RGB values are at three corners; cyan, magenta, and yellow are at three other corners; black is at the origin; and white is at the corner farthest from the origin.



RGB 24-bit color cube



The different colors in this space are points on or inside the cube, and are defined by vectors extending from the origin.

In this space, the gray scale (points of equal RGB values) extends from black to white along the line joining these two points.

RGB

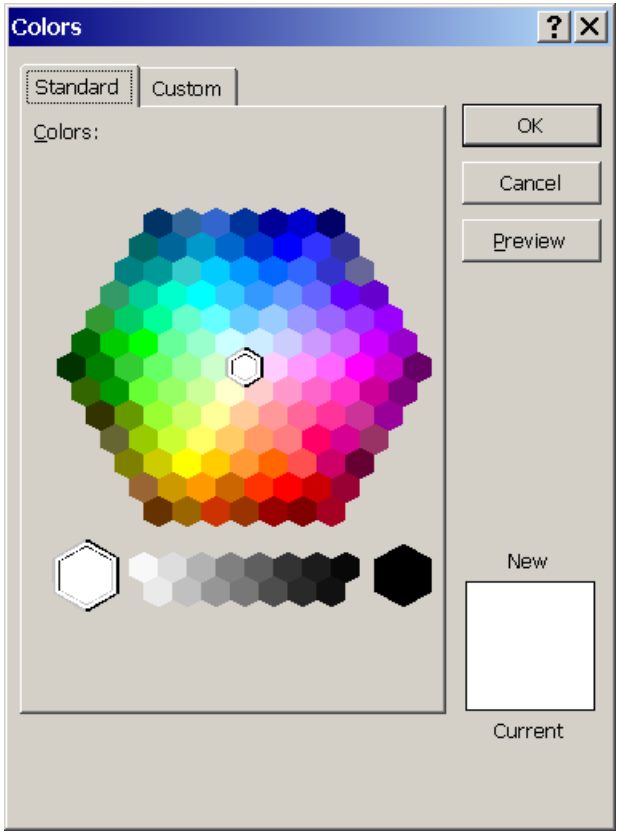
The RGB Color Space

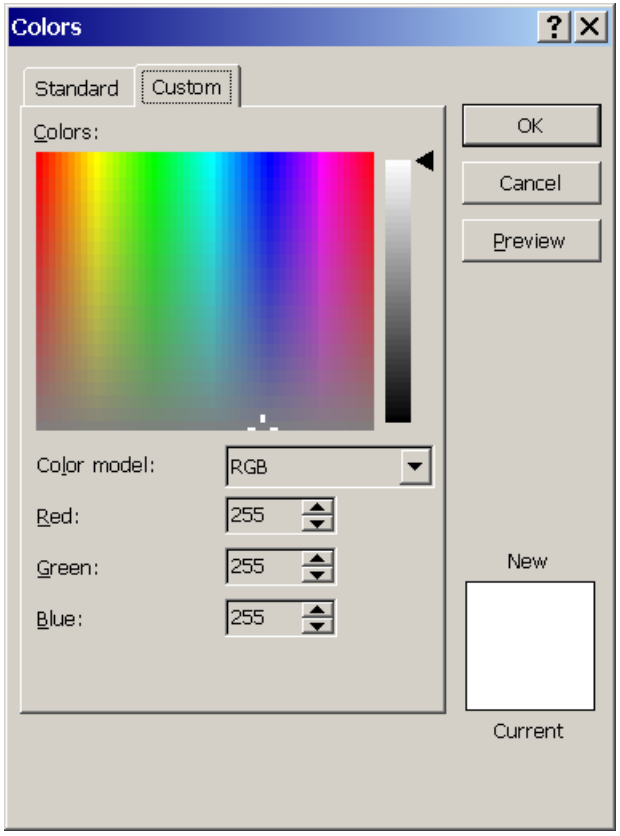
- 256 is a very convenient number to use in a digital representation, since a single 8-bit byte can hold exactly that many different values, usually considered as numbers in the range 0 to 255. Thus, an RGB color can be represented in three bytes, or 24 bits.

- The number of bits used to hold a color value is often referred to as the *color depth*.

- The common color depths are sometimes distinguished by the terms *millions of colors* (24 bit), *thousands of colors* (16 bit) and *256 colors* (8 bit).

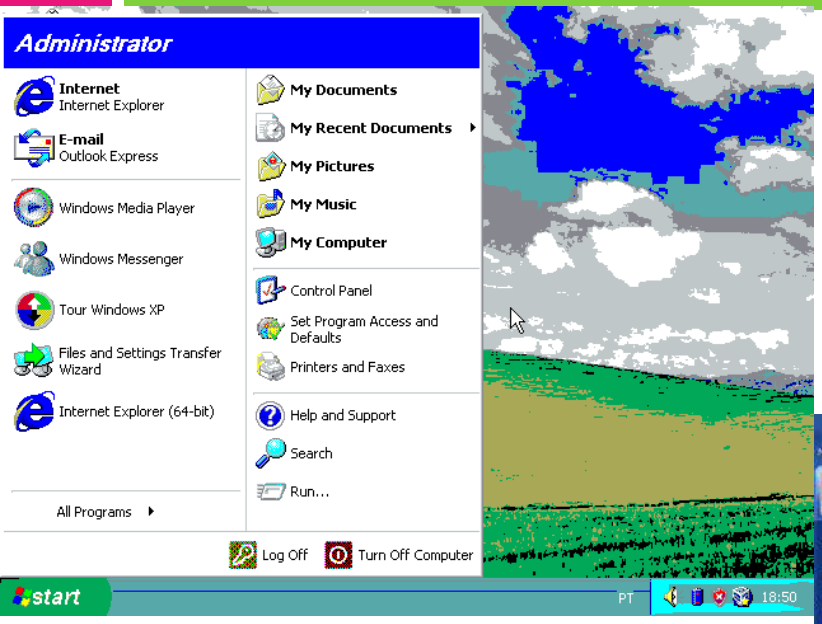
RGB





Color Selection

Color Depths



HSV

The HSV Color Space

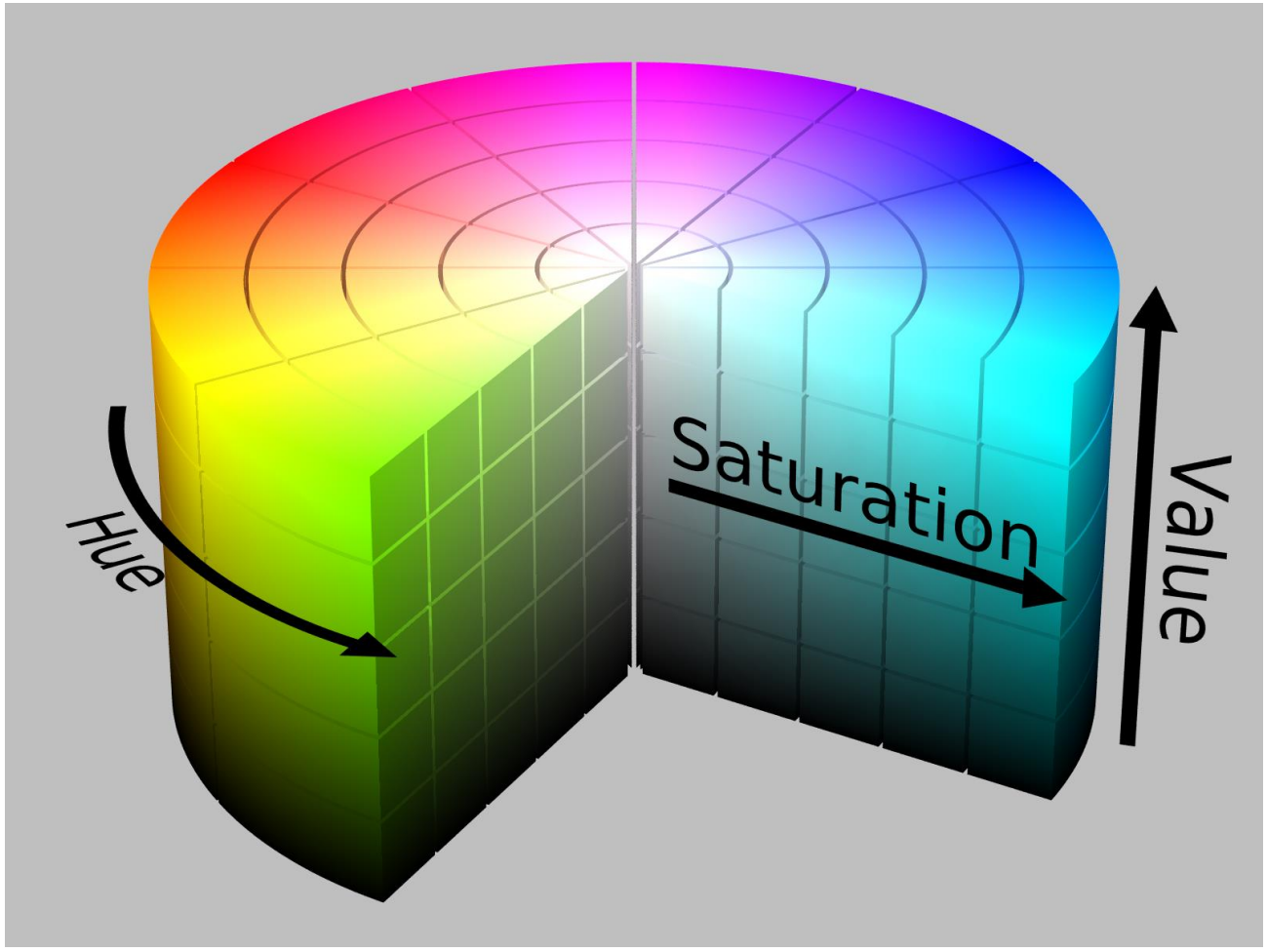
A particularly useful alternative method for representing (and manipulating) the colors of an image is known as the HSV color space. “HSV” refers to the hue, saturation and value of a pixel*.

In many ways, HSV space is a much more intuitive method of dealing with color, since it uses terms that match more closely with the way a layperson **talks about** color. When speaking of color conversationally, instead of characterizing a color as having ***85% red, 0% green, and 90% blue***, we would tend to say that the color is a “***saturated magenta”.*** The HSV model follows this thinking process, while still giving the user precise definition and control.

---

* Variations on the HSV model include HSI and HSB, in which the third component is either Intensity or Brightness, respectively.

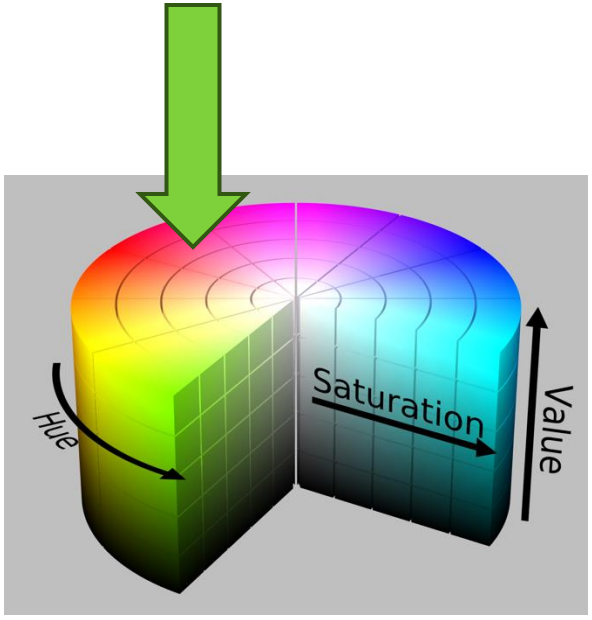
HSV(hue, saturation, and value)

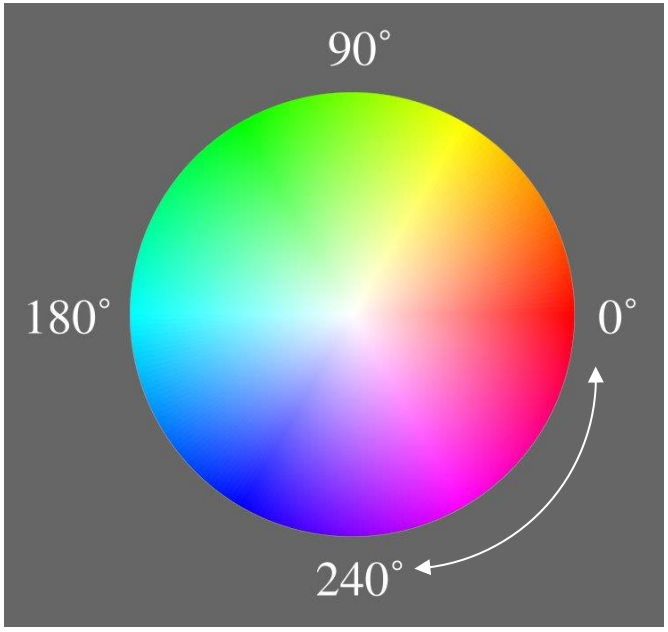


HSV

The HSV Color Model

The ***hue*** of a pixel refers to its basic color – such as red or yellow or violet or magenta. It is usually represented in the range of 0 to 360, referring to the color's location (in degree) around a circular color palette. For example, the color located at 90° corresponds to a yellow green, and pure blue is located at exactly 240° .

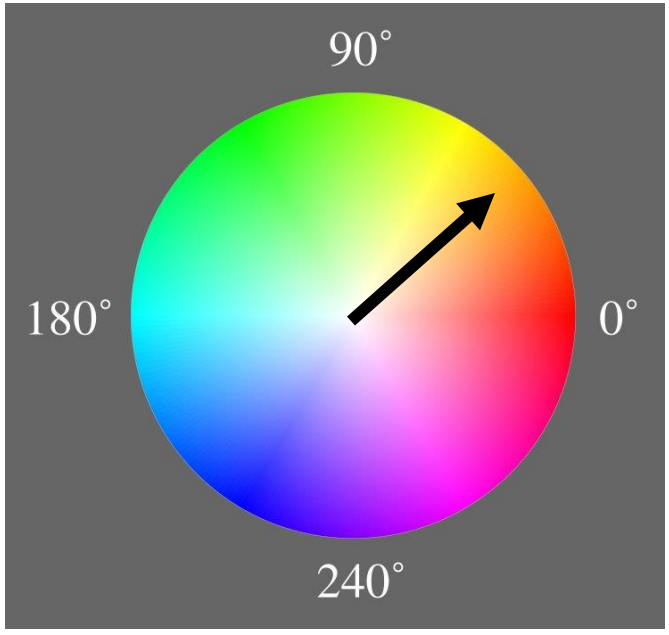




HSV

The HSV Color Model

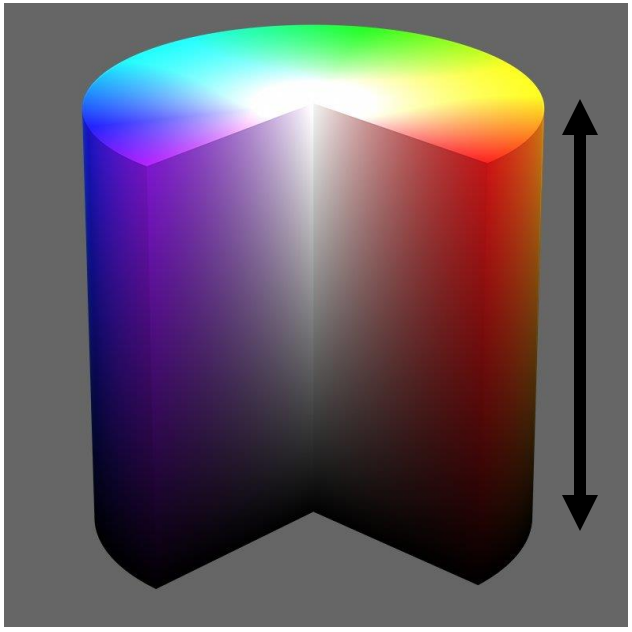
Saturation is the brilliance or purity of the specific hue that is present in the pixel. If we look again at the HSV color wheel, colors on the perimeter are fully saturated, and the saturation decreases as you move to the center of the wheel.



HSV

The HSV Color Model

Value can just be thought of as the brightness of the color, although strictly speaking it is defined to be the maximum of red, green, or blue values. Trying to represent this third component means that we need to move beyond a 2D graph. The value is graphed along the third axis, with the lowest value, black, being located at the bottom of the cylinder. White, the highest brightness value, is consequently located at the opposite end.

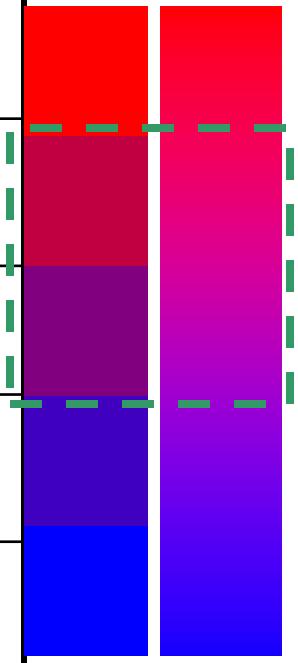


RGB → HSV

```python
def rgb_to_hsv(r, g, b):
    r, g, b = r / 255.0, g / 255.0, b / 255.0
    max_rgb = max(r, g, b)
    min_rgb = min(r, g, b)
    delta = max_rgb - min_rgb

    # Hue calculation
    if delta == 0:
        h = 0
    elif max_rgb == r:
        h = (60 * ((g - b) / delta) + 360) % 360
    elif max_rgb == g:
        h = (60 * ((b - r) / delta) + 120) % 360
    else:  # max_rgb == b
        h = (60 * ((r - g) / delta) + 240) % 360

    # Saturation calculation
    s = 0 if max_rgb == 0 else (delta / max_rgb)

    # Value calculation
    v = max_rgb

    return h, s * 100, v * 100

# Example usage:
rgb = (255, 128, 64)
hsv = rgb_to_hsv(*rgb)

print(f"RGB: {rgb} → HSV: {hsv}")
```

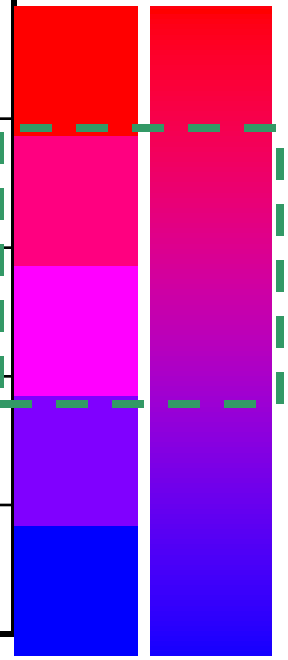
Color Interpolation Using RGB Colors

Fractional Time	HSV Colour	RGB Colour	Description
0.00		**1.0, 0.0, 0.0**	Bright red
0.25		0.75, 0.0, 0.25	Dark red-blue
0.50		0.5, 0.0, 0.50	Medium gray
0.75		0.25, 0.0, 0.75	Dark blue-red
1.0		**0.0, 0.0, 1.00**	Bright blue



Color Interpolation Using HSV Colors

Fractional Time	HSV Colour	RGB Colour	Description
0.00	**1.0, 0.0, 0.0**	**1.0, 0.0, 0.0**	Bright red
0.25	0.92, 0.0, 0.0	1.0, 0.0, 0.5	Bright red-blue
0.50	0.83, 0.0, 0.0	1.0, 0.0, 1.0	Bright purple
0.75	0.75, 0.0, 0.0	0.5, 0.0, 1.0	Bright blue-red
1.0	**0.67, 0.0, 0.0**	**0.0, 0.0, 1.0**	Bright blue



Comparison of Color Spaces

- RGB is most widely used in image storage and displaying.
- RGB is not **perceptually** uniform.

- CIE consists of luminance component and two chromatic components which are more perceptually attractive.

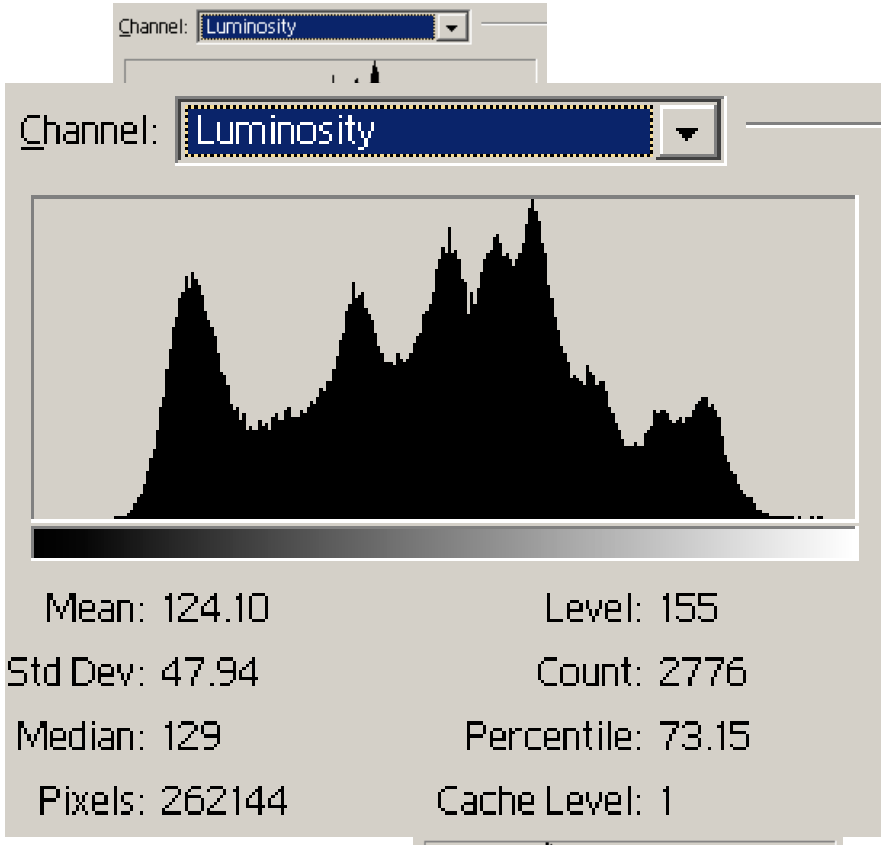
- HSV is perceptually uniform.
- HSV is widely used in computer graphics and is often selected due to its invariant properties of illumination and camera direction.

There is no agreement on which is the best choice.

Color Histogram

- A distribution of colors.
- The color histogram serves as an effective representation of the color content of an image if the color pattern is unique compared with the rest of the data set.
- The color histogram is easy to compute and effective in characterizing both the global and local distributions of colors in an image.
- It is also **robust to translation and rotation** about the viewing axis and changes only slowly with the scale and viewing angle.
- Since any pixel in the image can be described by three components in a certain color space, a histogram, i.e., the distribution of the number of pixels for each quantized bin, can be defined for each color component.

Color Histogram

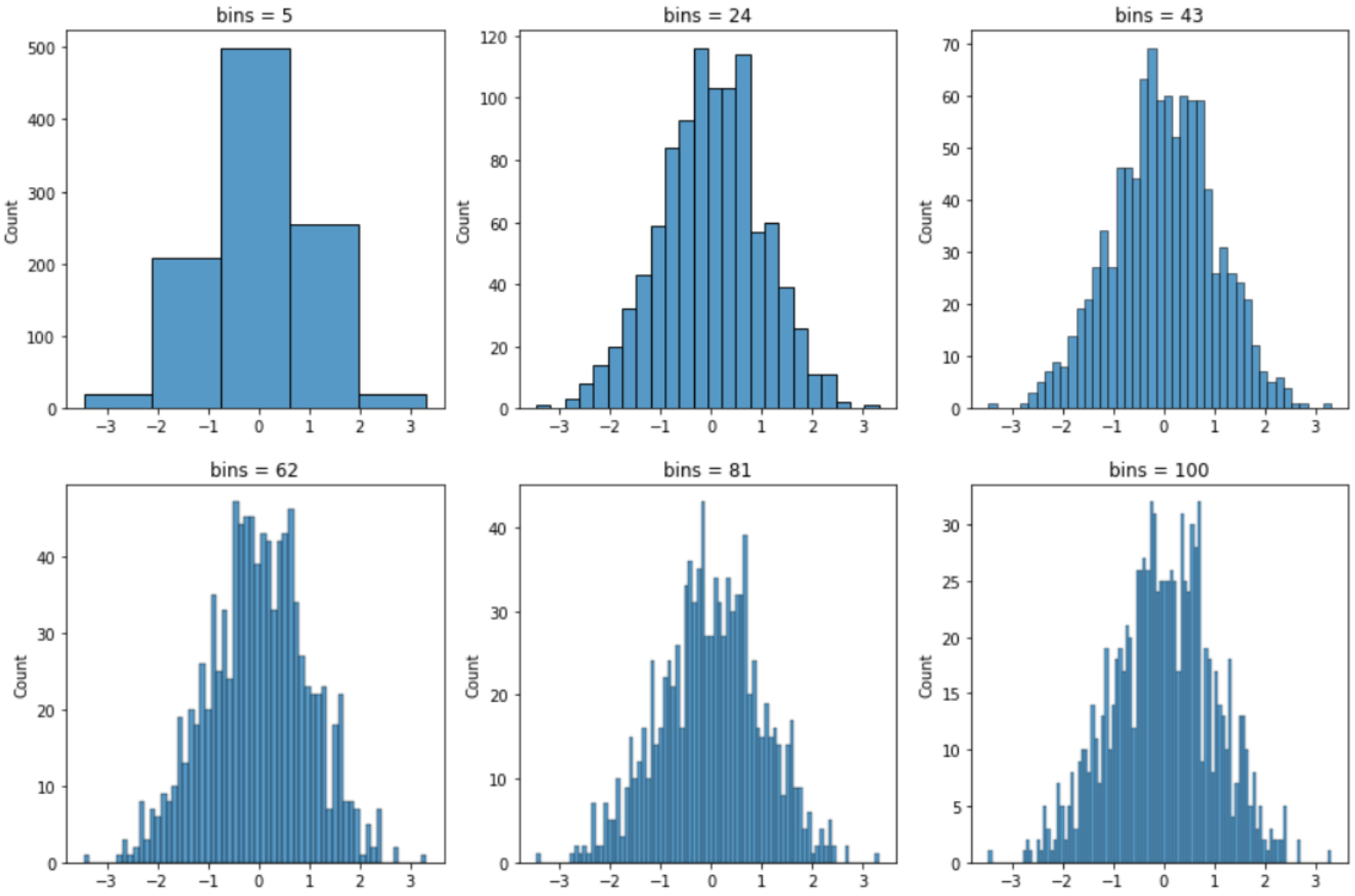


$$F_i(I) = \frac{\text{The number of pixels with color } i}{\text{The number of pixels within image } I}$$

Color Histogram

- The more bins a color histogram contains, the more discrimination power it has. However, a histogram with a large number of bins will not only increase the computational cost but will also be inappropriate for building efficient indexes for images databases.
- Furthermore, a very fine bin quantization does not necessarily improve the retrieval performance in many applications.
- Need to reduce the number of bins
- Essentially a density estimation

Color Histogram: Bin number



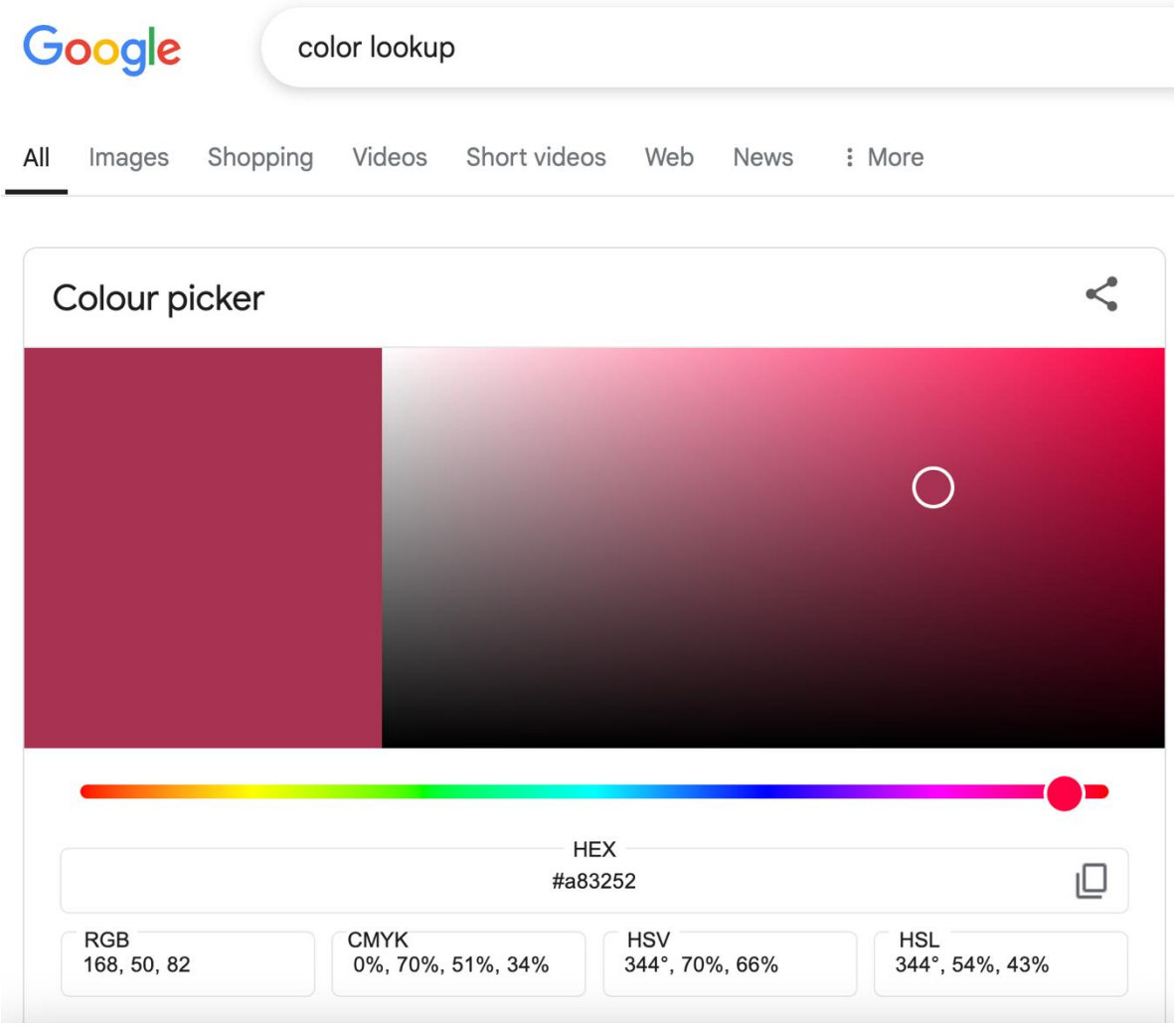
- https://medium.datadriveninvestor.com/how-to-decide-on-the-number-of-bins-of-a-histogram-3c36dc5b1cd8

Color Look-up Table

Color	R	G	B
Black	0	0	0
Dar Blue	0	0	128
Blue	0	0	255
Dark Green	0	128	0
Turquoise	0	128	128
Sky Blue	0	128	255
Green	0	255	0
Spring Green	0	255	128
Cyan	0	255	255
Brown	128	0	0
Violet	128	0	255
Marine Blue	128	0	255
Olive Drab	128	128	0
Gray	128	128	128

Color	R	G	B
Slate Blue	128	128	255
Lawn Green	128	255	0
Pale Green	128	255	128
Light Cyan	128	255	255
Red	255	0	0
Maroon	255	0	128
Magenta	255	0	255
Orange	255	128	0
Pink	255	128	128
Light Magenta	255	128	255
Yellow	255	255	0
Light Yellow	255	255	128
White	255	255	255

Color Look-up Table(Tools)



Color Histogram

- Reducing the number of bins –
  - Down-sampling the color depth / Quantization of the color space;
  - Use the bins that have the largest pixel numbers (since a small number of histogram bins capture the majority of pixels of an image). Such a reduction does not degrade the performance of histogram matching, but may even enhance it since small histogram bins are likely to be noisy;
  - Clustering methods – determine the *K* best colors and then calculate the number of pixels that fall in each of the *K* best colors.
  - Semantic description such as color space of X11 systems.

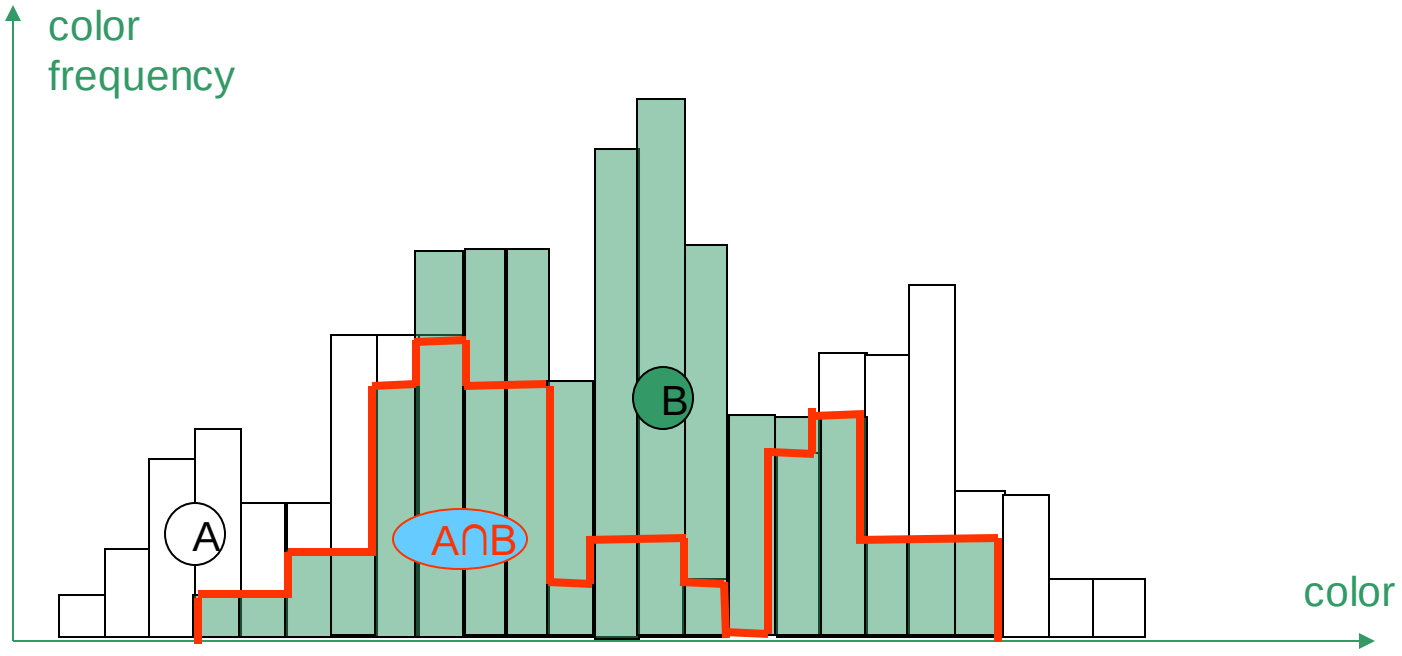
Histogram Intersection

- Histogram Intersection is employed to measure the similarity between two histograms.

$$F_i(I) = \frac{\text{The number of pixels with color } i}{\text{The number of pixels within image } I}$$

$$S(I_Q, I_D) = \frac{\sum_{i=1}^{N} \min(F_i(I_Q), F_i(I_D))}{\sum_{i=1}^{N} F_i(I_D)}$$

Histogram Intersection



- Colors that are not present in the query image do not contribute to the intersection distance.

Color Moments

- Color moments have been proved to be efficient and effective in representing color distributions of images, and have been successfully used in many retrieval systems (like IBM's QBIC), especially when the image contains only objects.
  - The first order moment (*mean)*

$$\mu_i = \frac{1}{N}\sum_{j=1}^{N} f_{ij}$$

  - The second order moment (*variance*)

$$\sigma_i = \left(\frac{1}{N}\sum_{j=1}^{N}(f_{ij} - \mu_i)^2\right)^{\frac{1}{2}}$$

  - The third order moment (*skewness*)

$$s_i = \left(\frac{1}{N}\sum_{j=1}^{N}(f_{ij} - \mu_i)^3\right)^{\frac{1}{3}}$$

where f_{ij} is the value of the i-th color component of the image pixel j, and N is the number of pixels in the image

Color Moments

- Since only 9 (three moments for each of the three color components) numbers are used to represent the color content of each image, color moments are very compact representations compared to other color features.

- Due to this compactness, they may also lower the discrimination power. Usually, color moments can be used as the first pass to narrow down the search space before other sophisticated color features are used for retrieval.

Color Coherence Vector

- Motivation
  - Color histogram does not present **spatial** information.
  - Color histogram is sensitive to both **compression artifacts** and camera **autogain**.
  - A special color descriptor taking spatial information into account should be proposed.
- Color coherence vector (CCV) can be used to distinguish images whose color histograms are indistinguishable.

Color Coherence Vector

Two images with similar color histogram, even their appearances are significantly different..

Their red colors!

G. Pass, R. Zabih, J. Miller

Color Coherence Vector

- A color's coherence is defined as the degree to which pixels of that color are members of large similarly-colored regions.
- These significant regions are referred as coherent regions which are observed to be of significant importance in characterizing images.
- Coherence measure classifies pixels as either coherent or incoherent.
- A color coherence vector represents this classification for each color in the image.

Computing CCV

- Conduct average filtering on the image.
  - To eliminate small variations between neighbour pixels.
- Discretize the image into *n* distinct colors.
- Classify the pixels within a give color bucket as either coherent or incoherent.
  - A pixel is coherent if the size of its connected component exceeds a fixed value τ; otherwise, the pixel is incoherent.
- Obtain CCV by collecting the information of both coherent and incoherent into a vector
  - CCV $[(\alpha_1,\beta_1), (\alpha_2,\beta_2), \ldots, (\alpha_n,\beta_n)]$, where α and β are the number of coherent pixels and incoherent pixels of the color respectively.

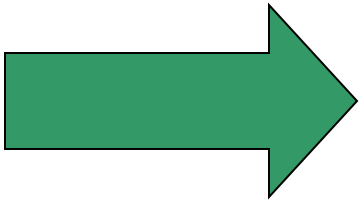
Computing CCV

$\tau = 4$, R=G=B

22	10	21	22	15	16
24	21	13	20	14	17
23	17	38	23	17	16
25	25	22	14	15	14
27	22	12	11	17	18
24	21	10	12	15	19

Blurred Image

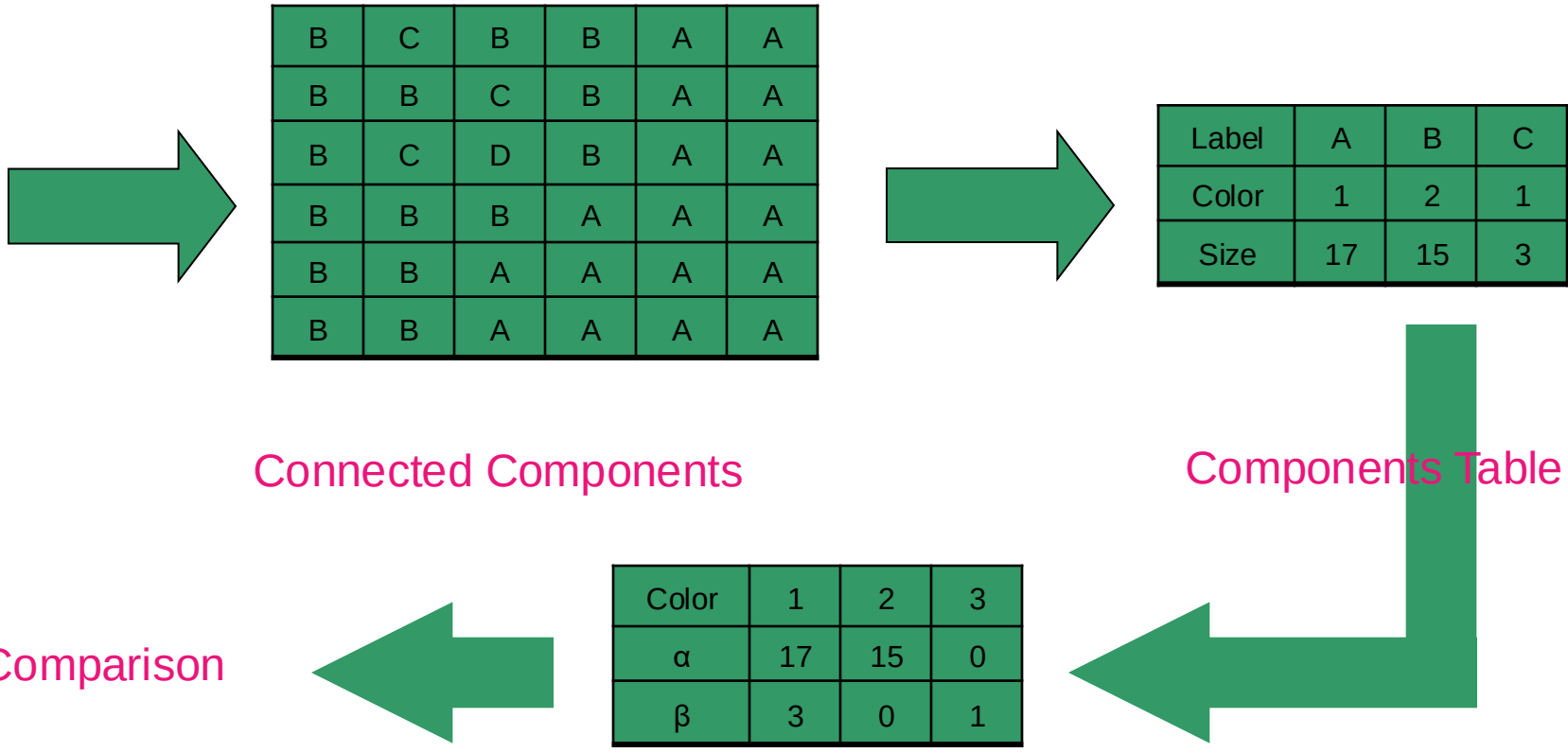
1: 10-19
2: 20-29
3: 30-39



2	1	2	2	1	1
2	2	1	2	1	1
2	1	3	2	1	1
2	2	2	1	1	1
2	2	1	1	1	1
2	2	1	1	1	1

Discretized Image

Computing CCV



Comparing CCVs

Non-normalized

$$dist = \sum_{i=1}^{n} \left| (\alpha_i - \alpha_i') \right| + \left| (\beta_i - \beta_i') \right|$$

Normalized

$$dist = \sum_{i=1}^{n} \left| \frac{\alpha_i - \alpha_i'}{\alpha_i + \alpha_i' + 1} \right| + \left| \frac{\beta_i - \beta_i'}{\beta_i + \beta_i' + 1} \right|$$

Histogram vs. CCV

Normalized

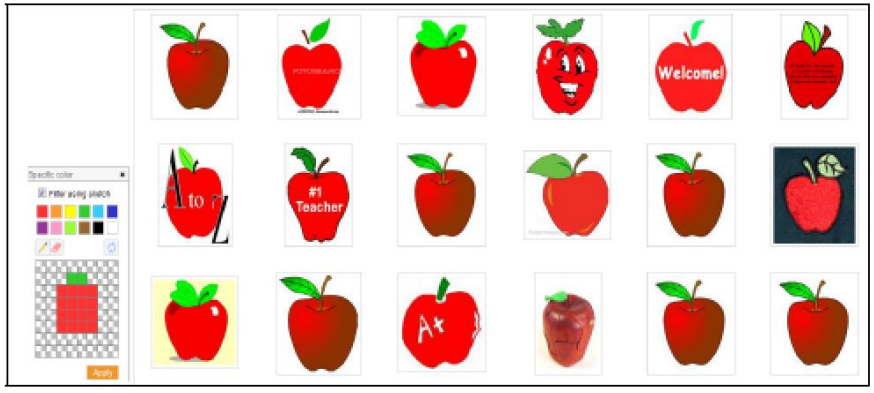
Histogram best match

CCV best match

G. Pass, R. Zabih, J. Miller
Cornell University

Image Search by Color Sketch

a)



b)

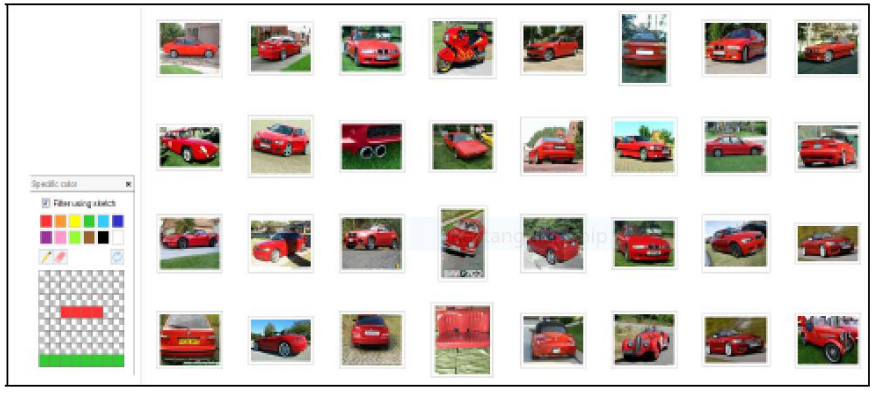
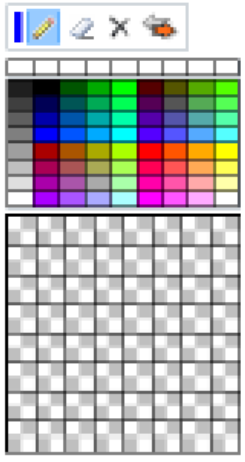
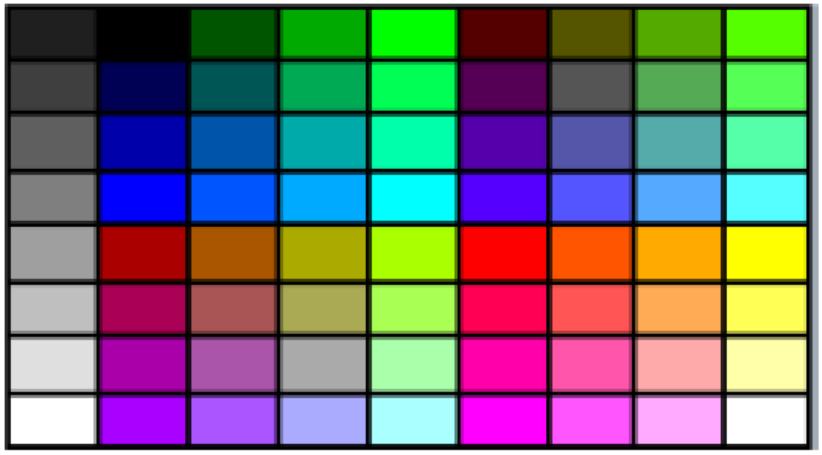


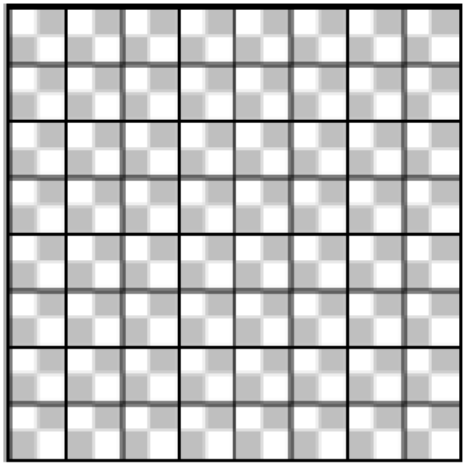
Image Search by Color Sketch



(a)



(b)

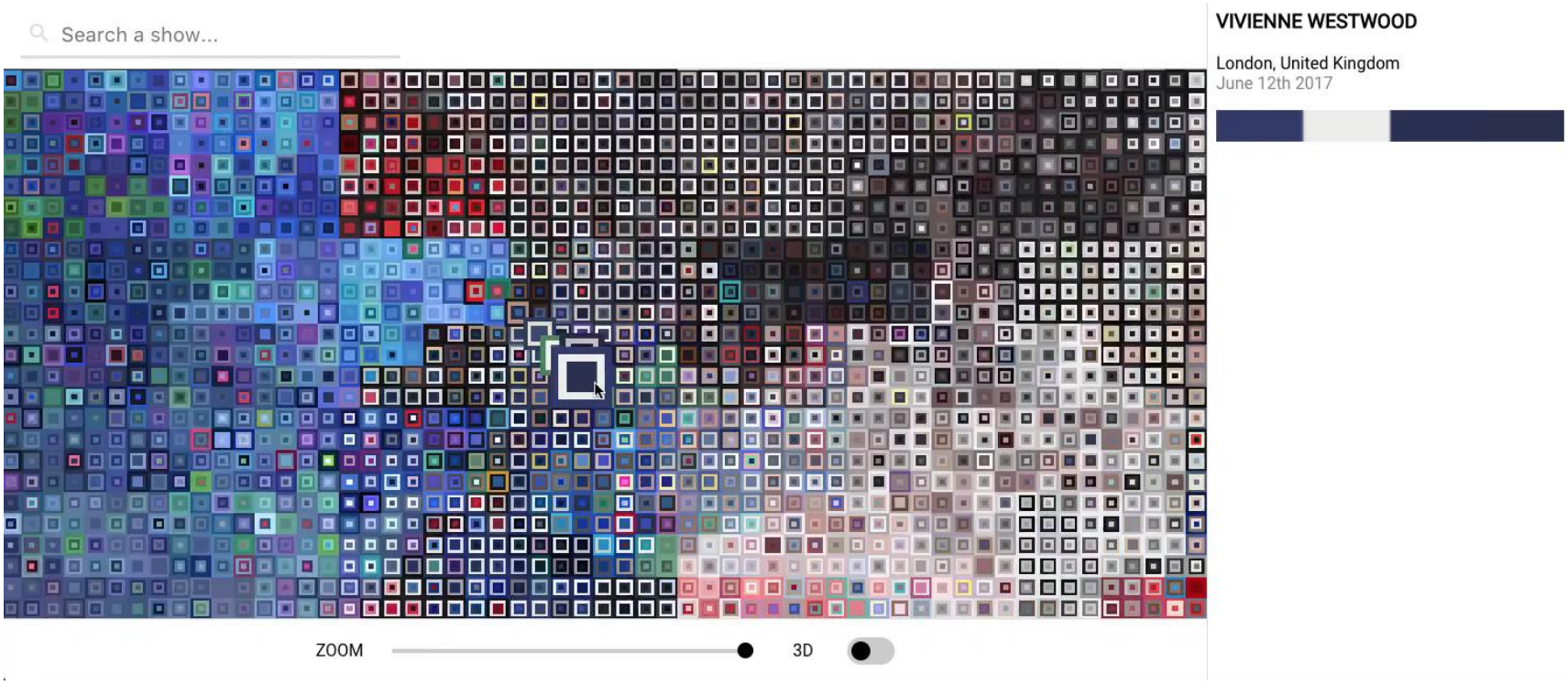


(c)

Image Search by Color

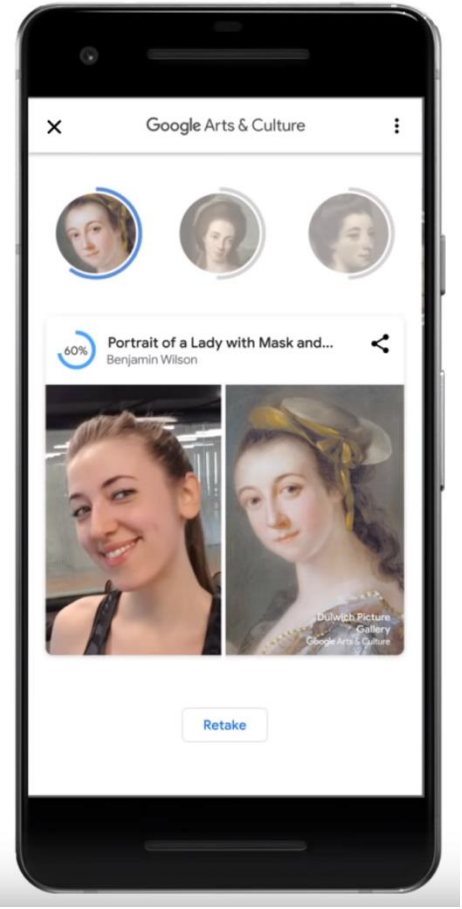
https://www.fastcompany.com/90166015/this-new-google-tool-is-like-reverse-image-search-for-color-palettes

Fashion and Color



https://experiments.withgoogle.com/business-of-fashion

Selfie and Art



https://blog.google/outreach-initiatives/arts-culture/exploring-art-through-selfies-google-arts-culture/

Texture

- "**Texture** - ...(*in extended use*) the constitution, structure, or substance of anything with regard to its constituents or formative elements." (The Oxford Dictionary, 1971; 1989).

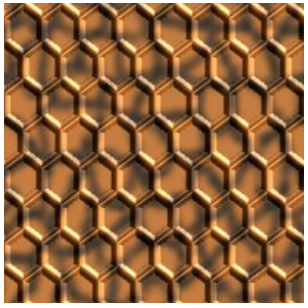
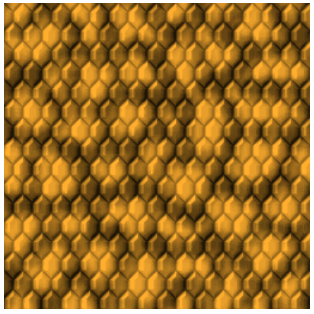
- "**Texture** - ...a basic scheme or structure; the overall structure of something incorporating all of most of parts." (Webster's Dictionary, 1959; 1986).

Texture

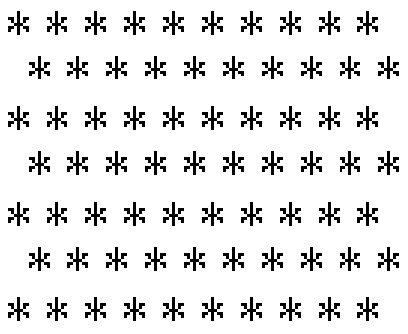
- The concept of texture is intuitively obvious but has no precise definition.
- Texture is something consisting of mutually related elements.
- Texture primitives (or texture elements, texels) are building blocks of a texture.
- Texture description is scale dependent.

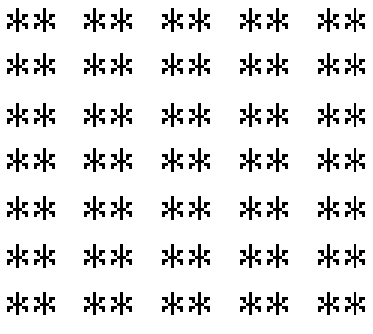
- Humans describe texture as fine, coarse, grained, smooth, etc.
- These description are imprecise and non-quantitative.

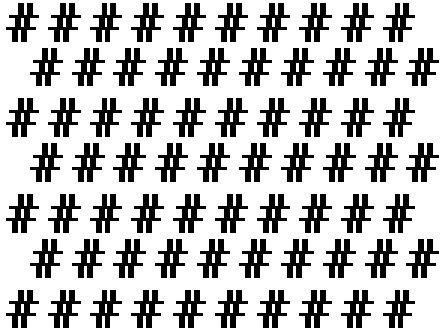
- Texture can be described by its tone and structure.
- Tone ... based on pixel intensity properties.
- Structure ... describes spatial relationships of primitives.
- Texture can be described by the number and types of primitives and by their spatial relationships.

Texture







Neither identical primitives nor identical structure is sufficient for texture description if considered alone

1st Order Statistics

image $f \rightarrow h_f$ histogram

- Obtain statistics of the histogram:

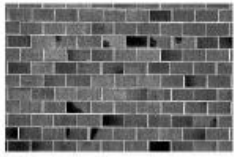
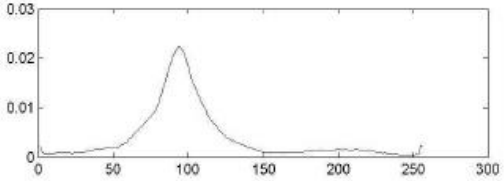
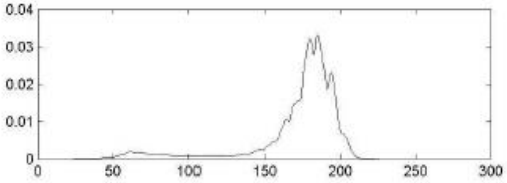
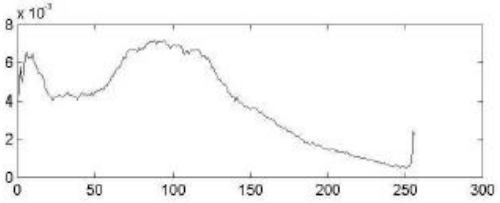
Mean: $\sum_{i=1}^{n} ih(i)$

Standard deviation: $\sum_{i=1}^{n} (i-\mu)^2 h(i)$

Skewness: $\sum_{i=1}^{n} (i-\mu)^3 h(i)$

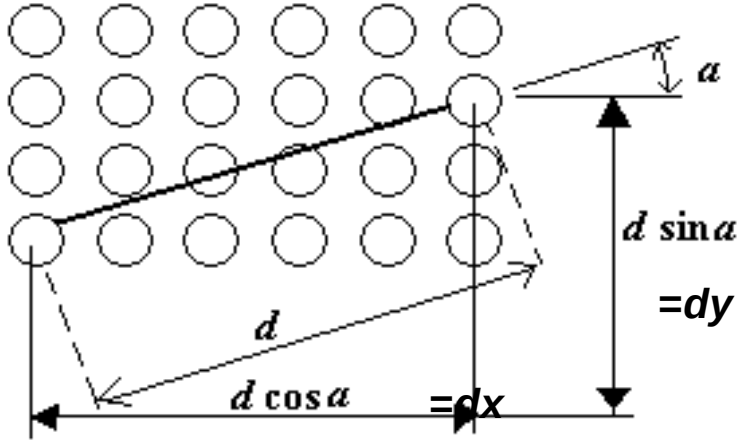
Entropy: $-\sum_{i=1}^{n} h(i) \log h(i)$ Measure of uniformity of histogram

1st Order Statistics

image			
histogram			
statistics	Skewness = 2.08 Entropy = 0.88	Skewness = 2.44 Entropy = 0.77	Skewness = -0.092 Entropy = 0.97

Co-occurrence Matrices

- Based on repeated occurrence of some gray-level configuration in the texture
- This configuration varies rapidly in fine textures, more slowly in coarse textures
- Occurrence of gray-level configuration may be described by matrices of relative frequencies, called co-occurrence matrices.
- A co-occurrence matrix shows how frequent is every particular pair of grey levels in the pixel pairs, separated by a certain distance *d* along a certain direction *a*.



Co-occurrence Matrices

0	0	1	1
0	0	1	1
0	2	2	2
2	2	3	3

$$
\begin{aligned}
P_{0°,d}(a,b) &= |\{((k,l),(m,n)) \in D : \\
&\quad k-m=0, |l-n|=d, \;\; f(k,l)=a, f(m,n)=b\}| \\
P_{45°,d}(a,b) &= |\{((k,l),(m,n)) \in D : \\
&\quad (k-m=d, l-n=-d) \;\; OR \;\; (k-m=-d, l-n=d), \\
&\quad f(k,l)=a, f(m,n)=b\}| \\
P_{90°,d}(a,b) &= |\{((k,l),(m,n)) \in D : \\
&\quad |k-m|=d, l-n=0, \;\; f(k,l)=a, f(m,n)=b\}| \\
P_{135°,d}(a,b) &= |\{((k,l),(m,n)) \in D : \\
&\quad (k-m=d, l-n=d) \;\; OR \;\; (k-m=-d, l-n=-d), \\
&\quad f(k,l)=a, f(m,n)=b\}|
\end{aligned}
$$

$$
P_{0°,1} = \begin{array}{c|cccc} & \mathbf{0} & \mathbf{1} & \mathbf{2} & \mathbf{3} \\ \hline \mathbf{0} & 4 & 2 & 1 & 0 \\ \mathbf{1} & 2 & 4 & 0 & 0 \\ \mathbf{2} & 1 & 0 & 6 & 1 \\ \mathbf{3} & 0 & 0 & 1 & 2 \end{array}
\qquad
P_{135°,1} = \begin{array}{cccc} 2 & 1 & 3 & 0 \\ 1 & 2 & 1 & 0 \\ 3 & 1 & 0 & 2 \\ 0 & 0 & 2 & 0 \end{array}
$$

Co-occurrence Matrices

- Texture classification can be based on criteria (features) derived from the co-occurrence matrices
  - Energy
  - Entropy
  - Maximum probability
  - Contrast
  - Inverse difference moment
  - Correlation

$$\sum_{a,b} P^2_{\phi,d}(a,b)$$

$$\sum_{a,b} P_{\phi,d}(a,b) \log_2 P_{\phi,d}(a,b)$$

$$\max_{a,b} P_{\phi,d}(a,b)$$

$$\sum_{a,b} |a-b|^\kappa P^\lambda_{\phi,d}(a,b)$$

$$\sum_{a,b;a\neq b} \frac{P^\lambda_{\phi,d}(a,b)}{|a-b|^\kappa}$$

$$\frac{\sum_{a,b}[(ab)P_{\phi,d}(a,b)] - \mu_x\mu_y}{\sigma_x\sigma_y}$$

$$\mu_x = \sum_a a \sum_b P_{\phi,d}(a,b) \qquad \sigma_x = \sum_a (a-\mu_x)^2 \sum_b P_{\phi,d}(a,b)$$

$$\mu_y = \sum_b b \sum_a P_{\phi,d}(a,b) \qquad \sigma_y = \sum_b (b-\mu_x)^2 \sum_a P_{\phi,d}(a,b)$$

Co-occurrence Matrices

- Co-occurrence matrices description uses 2nd order image statistics.
- Advantage: consider spatial properties.
- Limitation: does not consider primitive shape.
- Therefore, not recommended for textures with large primitives.

Tamura Features

- Coarseness
- Contrast
- Directionality
- Linelikeness
- Regularity
- Roughness

Tamura Features : Coarseness

- Coarseness is a measure of the granularity of the texture.

$$A_k(x,y) = \sum_{i=x-2^{k-1}}^{x+2^{k-1}-1} \sum_{j=y-2^{k-1}}^{y+2^{k-1}-1} g(i,j)/2^{2k}$$

$$E_{k,h}(x,y) = \left| A_k(x+2^{k-1},y) - A_k(x-2^{k-1},y) \right|$$

$$E_{k,v}(x,y) = \left| A_k(x,y+2^{k-1}) - A_k(x,y-2^{k-1}) \right|$$

$$S_{best}(x,y) = 2^m$$

The value of m that maximizes E in either direction

$$F_{crs} = \frac{1}{m \times n} \sum_{i=1}^{m} \sum_{j=1}^{n} S_{best}(i,j)$$

Tamura Features : Contrast

- Contrast measures how grey levels vary in the image and to what extent their distribution is biased to black or white.

$$F_{con} = \frac{\sigma}{\alpha_4^n}$$

$\alpha_4 = \mu_4/\delta^4$, μ_4 is the 4th moment about the mean, δ^2 is the variance, $n = 0.25$ is recommended for the best of discriminating textures.

Tamura Features : Directionality

- Directionality is measured using the frequency distribution of oriented local edges against their directional angles.

Tamura Features : Directionality

- Computing directionality
  - Convolute image with two 3X3 arrays

$$\begin{matrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{matrix} \qquad \begin{matrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{matrix}$$

  - Compute a gradient vector (Δ_h, Δ_v) for each pixel.
  - Quantize θ and construct a histogram H_D of θ
  - Compute directionality by summarizing H_D:

$$|\Delta G| = \left(|\Delta_H| + |\Delta_V|\right)/2$$

$$\theta = \tan^{-1}\left(\Delta_V / \Delta_H\right) + \pi/2$$

$$F_{dir} = \sum_{p}^{n_p} \sum_{\phi \in w_p} \left(\phi - \phi_p\right)^2 H_D(\phi)$$

Where P ranges over n_p peaks, and for each peak p, w_p is the set of bins distributed over it.

H_D will present strong peaks for highly directional images.

Tamura Features

- Three other features are highly correlated with the above three features and do not add much to the effectiveness of the texture description.
- Linelikeness is defined as an average coincidence of the edge directions (more precisely, coded directional angles) that co-occurred in the pairs of pixels separated by a distance *d* along the edge direction in every pixel.
- Regularity $F_{reg}=1-r(s_{crs}+s_{con}+s_{dir}+s_{lin})$, s... means the standard deviation of the corresponding feature F... in each subimage the texture is partitioned into.
- Roughness $F_{rgh}=F_{crs}+F_{con}$
- These features capture the high-level perceptual attributes of a texture well and are useful for image browsing. However, they are not very effective for finer texture discrimination.

Gabor Filter Features

- As an alternative to the spatial domain for computing the texture features, Gabor filter features are multiresolution features obtained with Gabor filtering. The features describe spatial distributions of oriented edges in the image at multiple scales.

- A 2D Gabor function is defined as

$$f(x,y)=\frac{1}{2\pi\sigma_x\sigma_y}\exp\left[-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right)\right]\cdot\exp\left[2\pi\sqrt{-1}(wx+uy)\right]$$

Gabor Filter Features

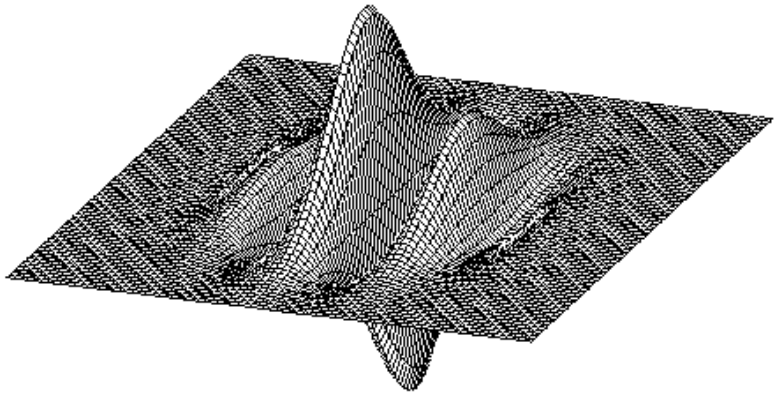
A set of Gabor filters can be obtained by appropriate dilations and rotations of $g(x, y)$

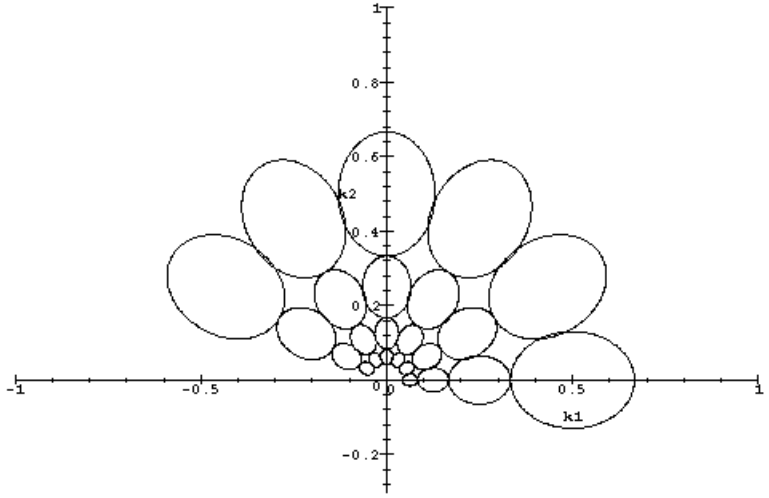
$$g_{mn}(x, y) = a^{-m} g(x', y')$$

$$x' = a^{-m}(x\cos\theta + y\sin\theta)$$

$$y' = a^{-m}(-x\sin\theta + y\cos\theta)$$

where $a > 1$, $\theta = n\pi/K$, $n = 0, 1, \ldots, K-1$, and $m = 0, 1, \ldots, S-1$. K and S are the number of orientations and scale

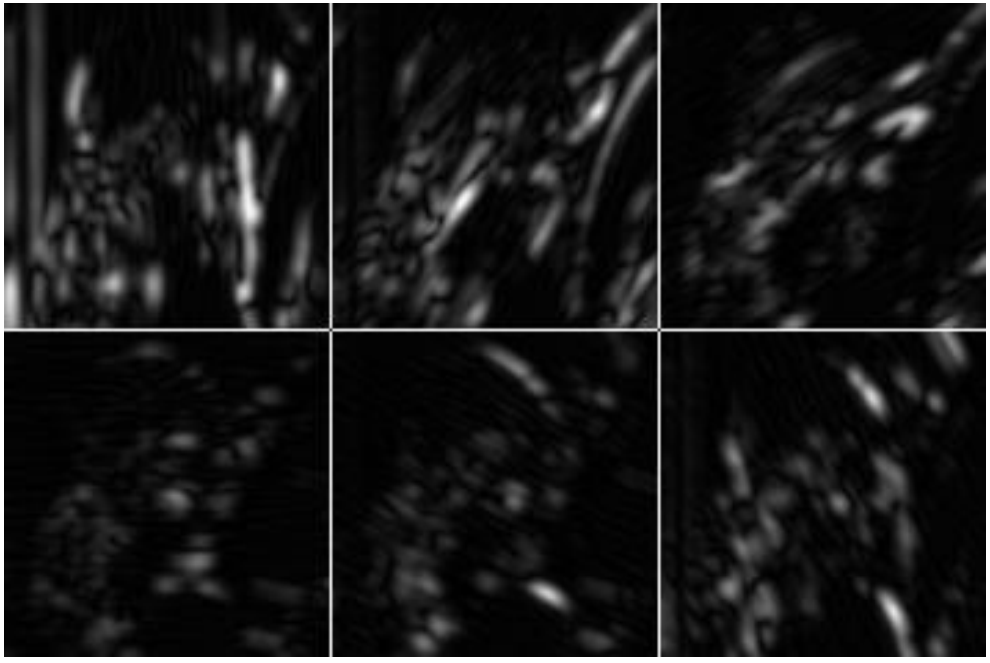




Gabor Filter Features

Gabor Filtering $W_{mn}(x,y) = I(x,y) * G_{mn}(x,y)$

$K = 6$ $S = 4$



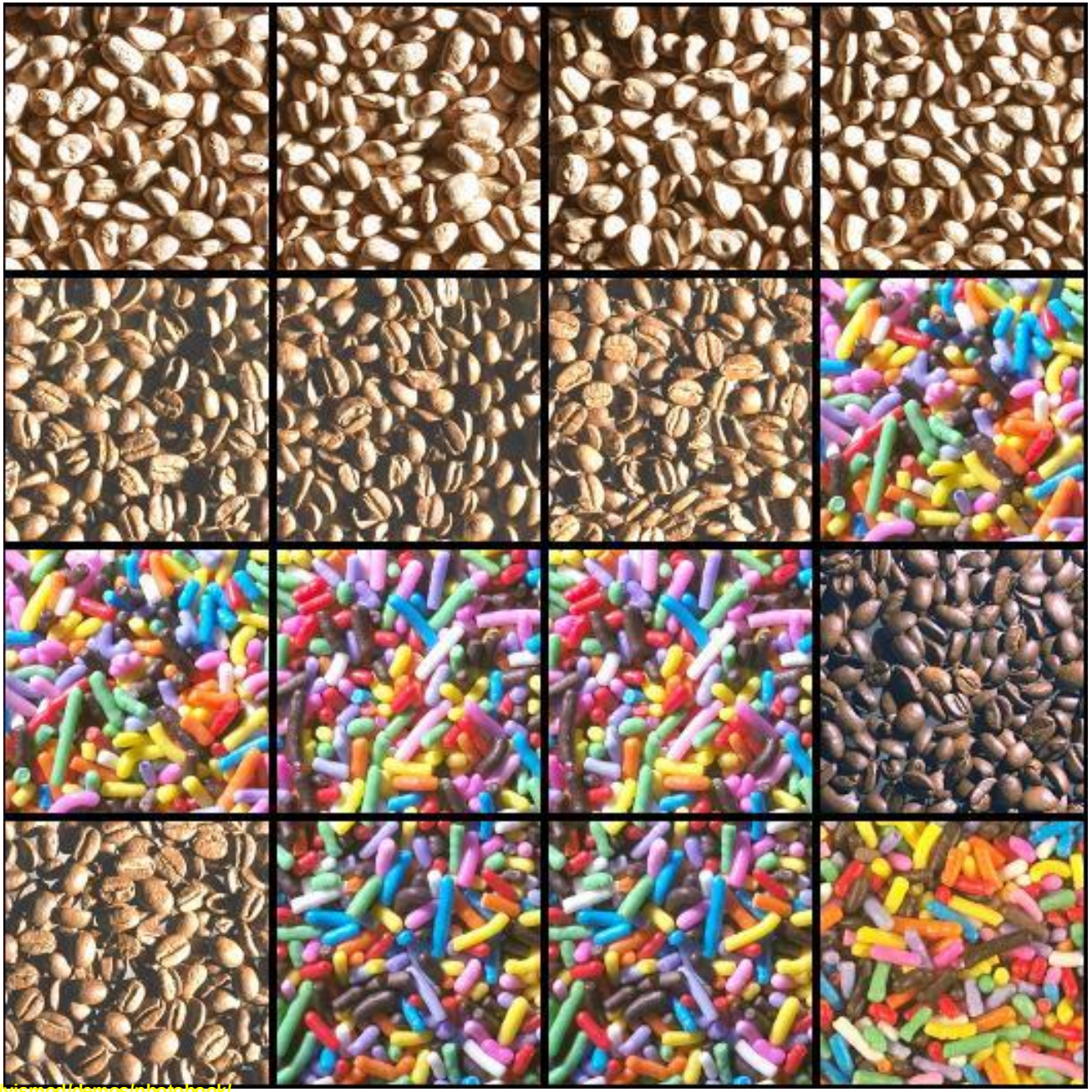
Gabor Features $[\mu_{00}, \delta_{00}, \ldots, \mu_{mn}, \delta_{mn}, \ldots, \mu_{S-1,K-1}, \delta_{S-1,K-1}]$

Texture

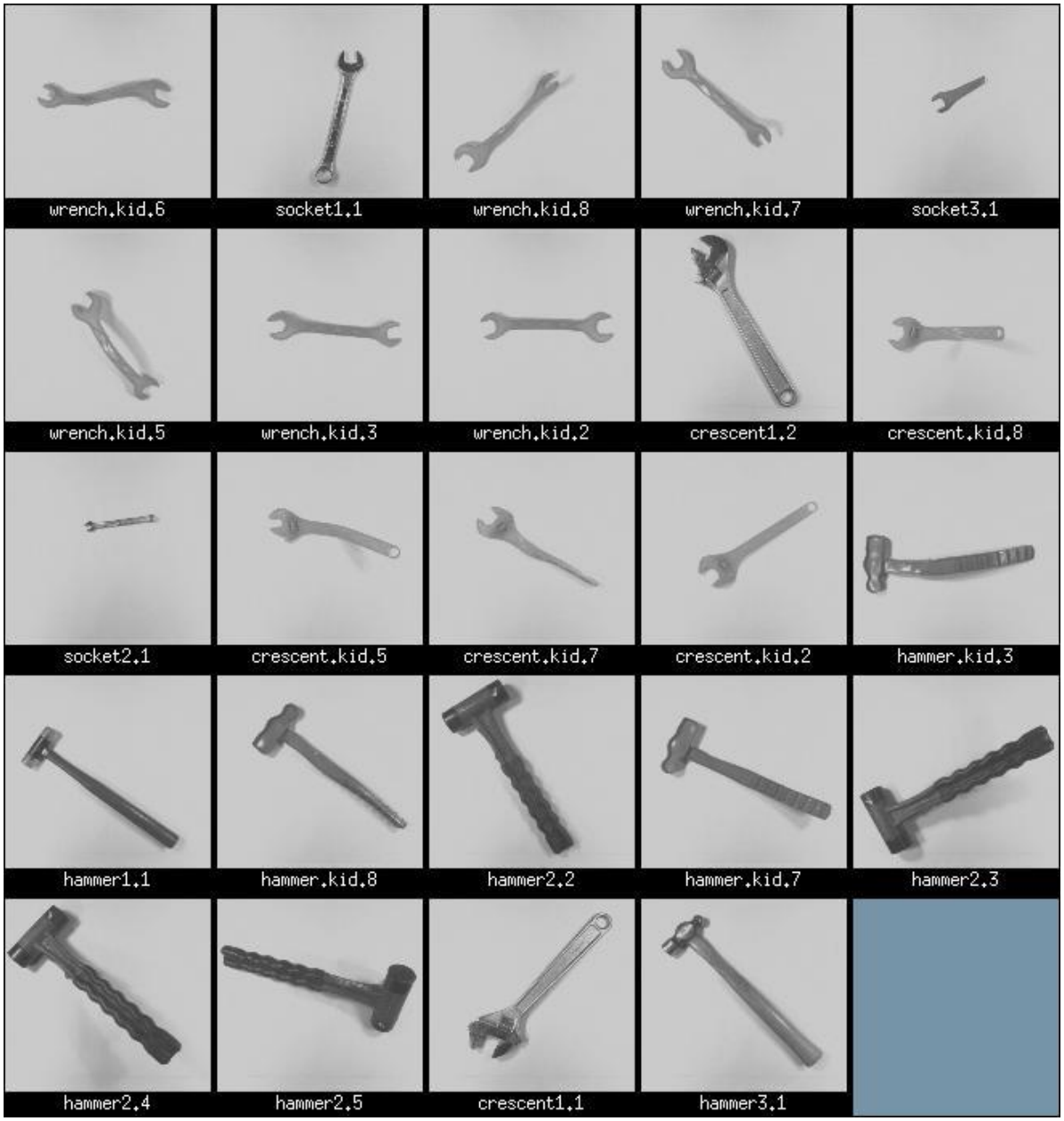
- Statistical texture descriptions, including Fourier Power Spectra, Co-occurrence Matrices, Edge Frequency, Shift-invariant Principal Component Analysis (SPCA), Tamura Feature, Wold Decomposition, Markov Random Field, Fractal Model, and multi-resolution filtering techniques such as Gabor and Wavelet transform, characterize texture by the statistical distribution of the image intensity.

- Structural texture descriptions, including Morphological Operator and Adjacency Graph methods, describe texture by identifying structural primitives and their placement rules. They tend to be most effective when applied to textures that are very regular.

- Hybrid texture descriptions.



http://vismod.media.mit.edu/vismod/demos/photobook/



http://vismod.media.mit.edu/vismod/demos/photobook/

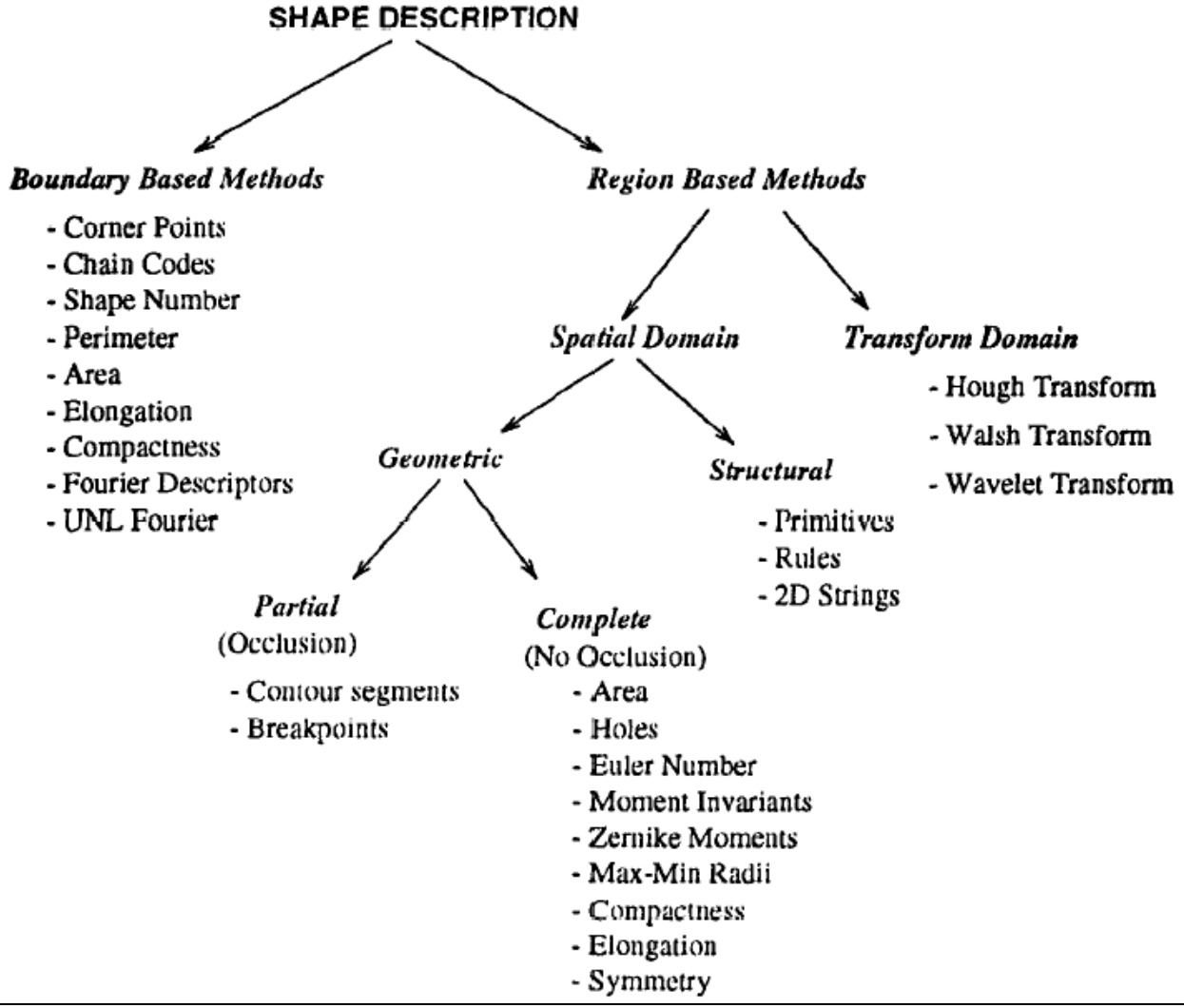
Shape

- The shape of an object or region refers to its profile and physical structure. Shape features of objects or regions have been used in many content-based image retrieval systems.

- Compared with color and texture features, shape features are usually described after images have been **segmented** into regions or objects.

- Since robust and accurate image segmentation is difficult to achieve, the use of shape features for image retrieval has been limited to special application where objects or regions are readily available.

Shape Representation

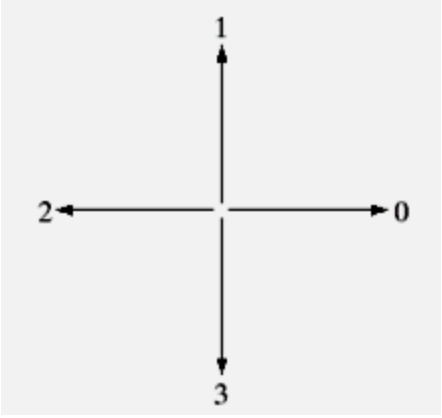


Combinations to better characterize shape patterns.

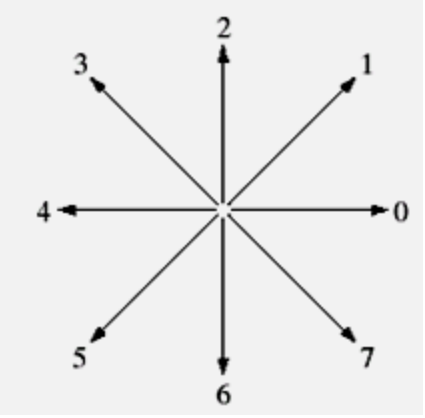
Chain Code

- Chain codes are used to represent a boundary by a connected sequence of straight-line segments of specified length and direction.

- Typically, this representation is based on 4- or 8-connectivity of the segments. The direction of each segment is coded by using a numbering scheme.

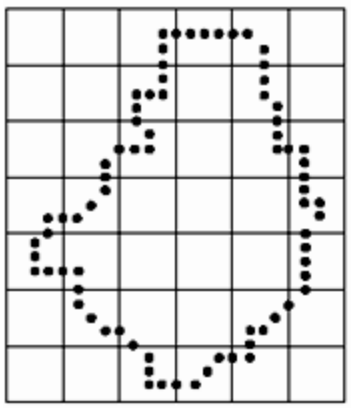


direction numbers for
4-directional chain code

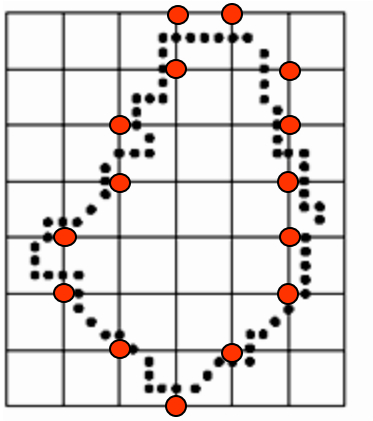


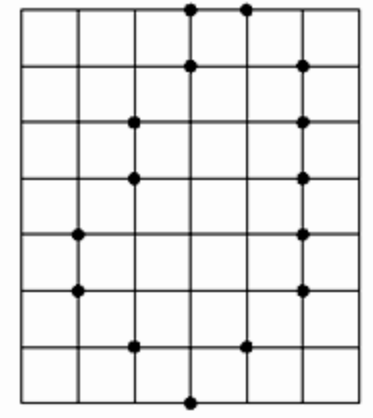
direction numbers for
8-directional chain code

Chain Code

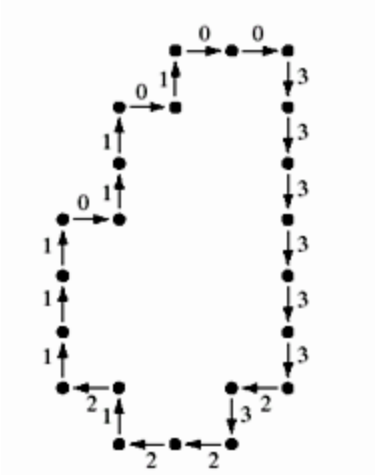


As the boundary is traversed, a boundary point is assigned to each node of the large grid, depending on the proximity of the original boundary to that node.

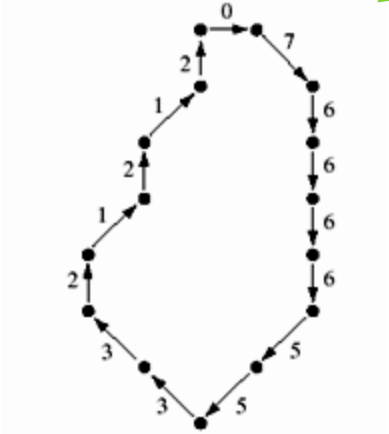




Resample the boundary by selecting a larger grid spacing

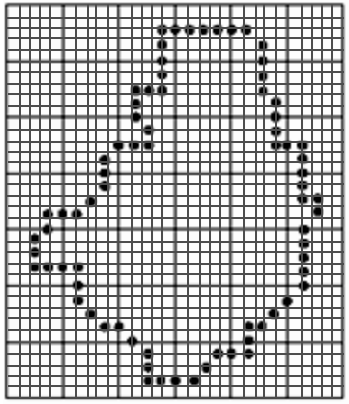


4-directional chain code



8-directional chain code

Chain Code

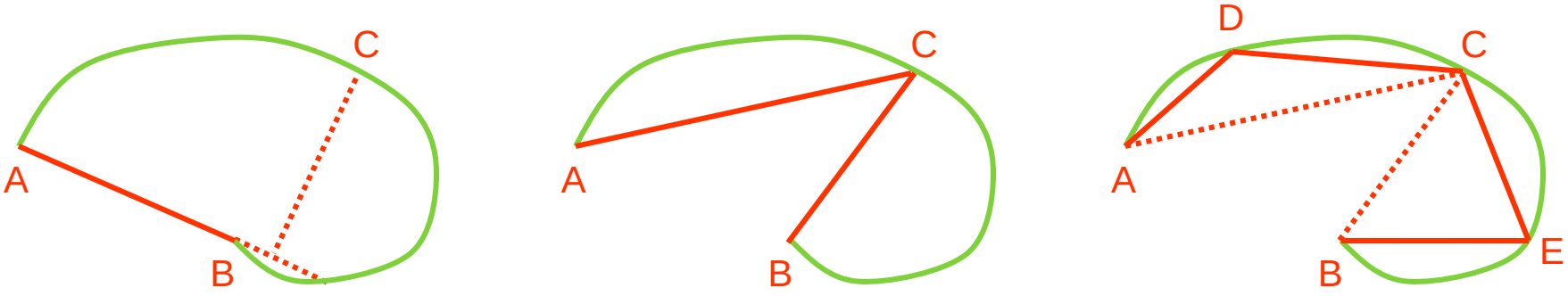


- Digital images usually are acquired and processed in a grid format with equal spacing in the x- and y-directions, so a chain code could be generated by following a boundary in a clockwise direction and assigning a direction to the segments connecting every pair of pixels.

- However, this method generally is unacceptable for two principal reasons:
  - The resulting chain of codes tends to be quite long;
  - Any small disturbances along the boundary due to noise or imperfect segmentation cause changes in the code that may not be related to the shape of the boundary.

Fitting Line Segments

- Straight-line segments can give simple approximation of curve boundaries. An interesting sequential algorithm for fitting a curve by line segments is as follows:



- Approximate the curve by the line segment joining its end points (A, B);
- If the distance from the farthest curve point (C) to the segment is greater than a predetermined quantity, join AC and BC;
- Repeat the procedure for new segments AC and BC, and continue until the desired accuracy is reached.

B-spline representation for curve approximation

Fourier Descriptors (FDs)

- Fourier Descriptors describe the object contour in the frequency domain.

$$u(l) = x(l) + jy(l)$$

$$a_n = \frac{1}{2\pi n} \sum_{k=1}^{N_V} (b_{k-1} - b_k) e^{-j(\frac{2\pi n l_k}{L})}$$

where L is the total length of curve $u(l)$, $l_k = \Sigma^k_{i=1}|V_i - V_{i-1}|$, and bk= $(Vk+1-Vk)/|V_{K-1}-V_k|$, N_v is the number of sample points, N_c is the number of FD coefficients used..

$$d(\alpha, \beta) = \sqrt{\sum_{n=-N_c}^{N_c} |a_n - b_n|^2}$$

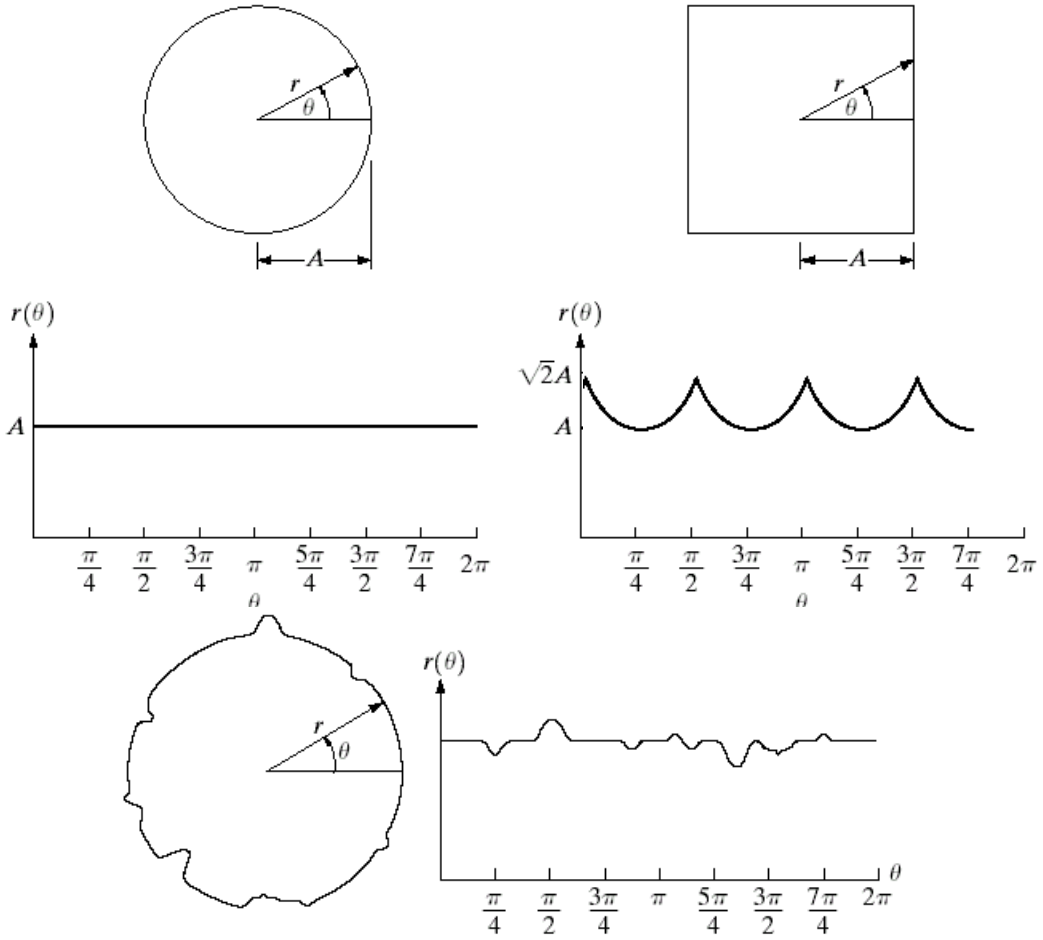
Fourier Descriptors (FDs)

- To account for the effects of scale (s), rotation (φ), and the starting point (p), we must minimize the distance metric

$$d^*(\alpha,\beta) = \min_{s,\phi,p} \sum_{n=-M, n\neq 0}^{M} \left| a_n - s e^{j(np+\phi)} b_n \right|^2$$

This is a computationally expensive optimization problem, which makes such an FD impractical for shape matching in real time.

Distance-angle Signature



- Translation invariance
- Rotation invariance
- Scaling invariance
- Robustness

This signature is also called Centroid-contour distance curve

Moments of 2D Function

- For a 2D continuous function $f(x, y)$, the moment of order ($p+q$) is defined as:

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dx dy \quad \text{(Eq. 1)}$$

where p , $q = 0, 1, 2, \ldots$. A uniqueness theorem states that if $f(x, y)$ is piecewise continuous and has nonzero values only in a finite part of the xy-plane, moments of all orders exist, and the moment sequence (m_{pq}) is uniquely determined by $f(x, y)$. Conversely, (m_{pq}) uniquely determines $f(x, y)$.

- The central moments are defined as:

$$\mu_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x - \bar{x})^p (y - \bar{y})^q f(x, y) dx dy \quad \text{(Eq. 2)}$$

where $\bar{x} = \dfrac{m_{10}}{m_{00}}$ and $\bar{y} = \dfrac{m_{01}}{m_{00}}$

Moments of Digital Image

- If $f(x, y)$ is a digital image, then Eq.2 becomes

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad \text{(Eq. 3)}$$

- The central moments of order up to 3 can then be obtained:

$$\mu_{00} = m_{00} \qquad \mu_{02} = m_{02} - \bar{y} m_{01}$$

$$\mu_{10} = 0 \qquad \mu_{30} = m_{30} - 3\bar{x} m_{20} + 2\bar{x}^2 m_{10}$$

$$\mu_{01} = 0 \qquad \mu_{03} = m_{03} - 3\bar{y} m_{02} + 2\bar{y}^2 m_{01}$$

$$\mu_{11} = m_{11} - \bar{y} m_{10} \qquad \mu_{21} = m_{21} - 2\bar{x} m_{11} - \bar{y} m_{20} + 2\bar{x}^2 m_{01}$$

$$\mu_{20} = m_{20} - \bar{x} m_{10} \qquad \mu_{12} = m_{12} - 2\bar{y} m_{11} - \bar{x} m_{02} + 2\bar{y}^2 m_{10}$$

Normalized Central Moments

- The normalized central moments, denoted η_{pq}, are defined as:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{r}} \quad \text{(Eq. 4)}$$

where

$$\gamma = \frac{p+q}{2} + 1$$

for $p+q = 2, 3, \ldots$.

Invariant Moments

- A set of seven invariant moments can be derived from the second-order and third-order moments:

$$\phi_1 = \eta_{20} + \eta_{02}$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$

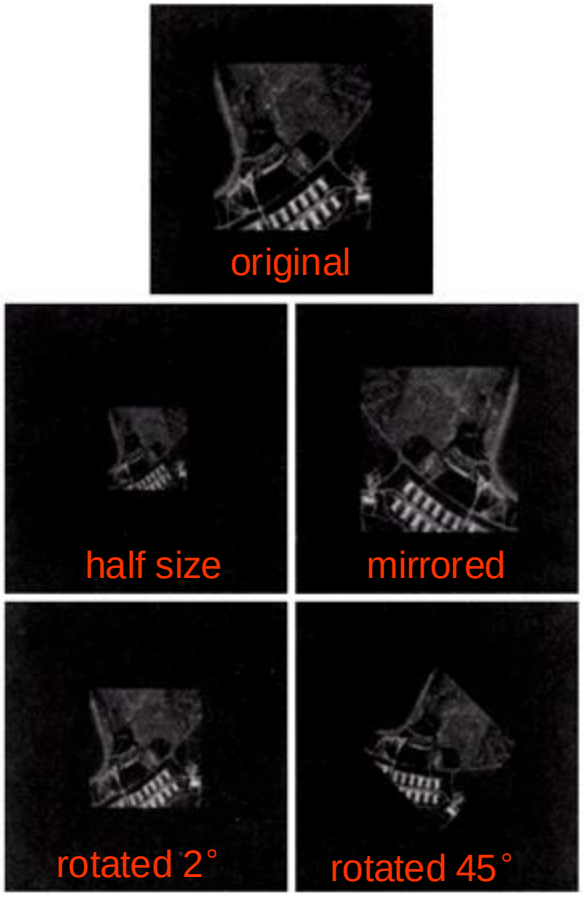
$$\phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right]$$

$$\phi_6 = (\eta_{20} - \eta_{02})\left[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$

$$\phi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right]$$

- This set of moments is invariant to translation, rotation, and scale change.

Invariant Moments

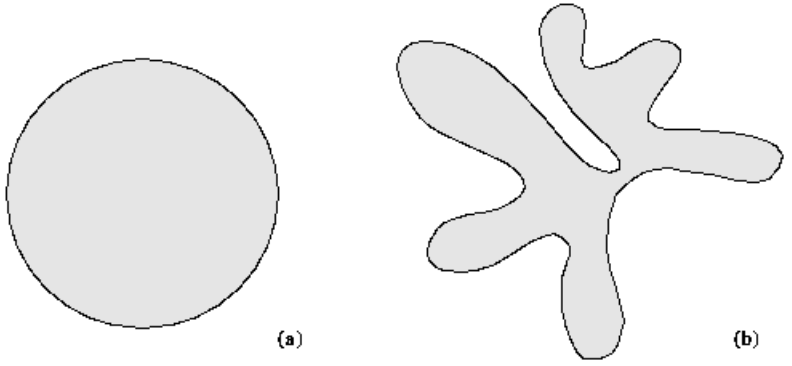


- The seven moment invariants were computed for each of these images, and the logarithm of the results were taken to reduce the dynamic range.

Invariant (Log)	Original	Half Size	Mirrored	Rotated 2°	Rotated 45°
ϕ_1	6.249	6.226	6.919	6.253	6.318
ϕ_2	17.180	16.954	19.955	17.270	16.803
ϕ_3	22.655	23.531	26.689	22.836	19.724
ϕ_4	22.919	24.236	26.901	23.130	20.437
ϕ_5	45.749	48.349	53.724	46.136	40.525
ϕ_6	31.830	32.916	37.134	32.068	29.315
ϕ_7	45.589	48.343	53.590	46.017	40.470

- The results for these images are in reasonable agreement with the invariants computed for the original image. The major cause of error can be attributed to the digital nature of the data, especially for the rotated images.

Geometry Features

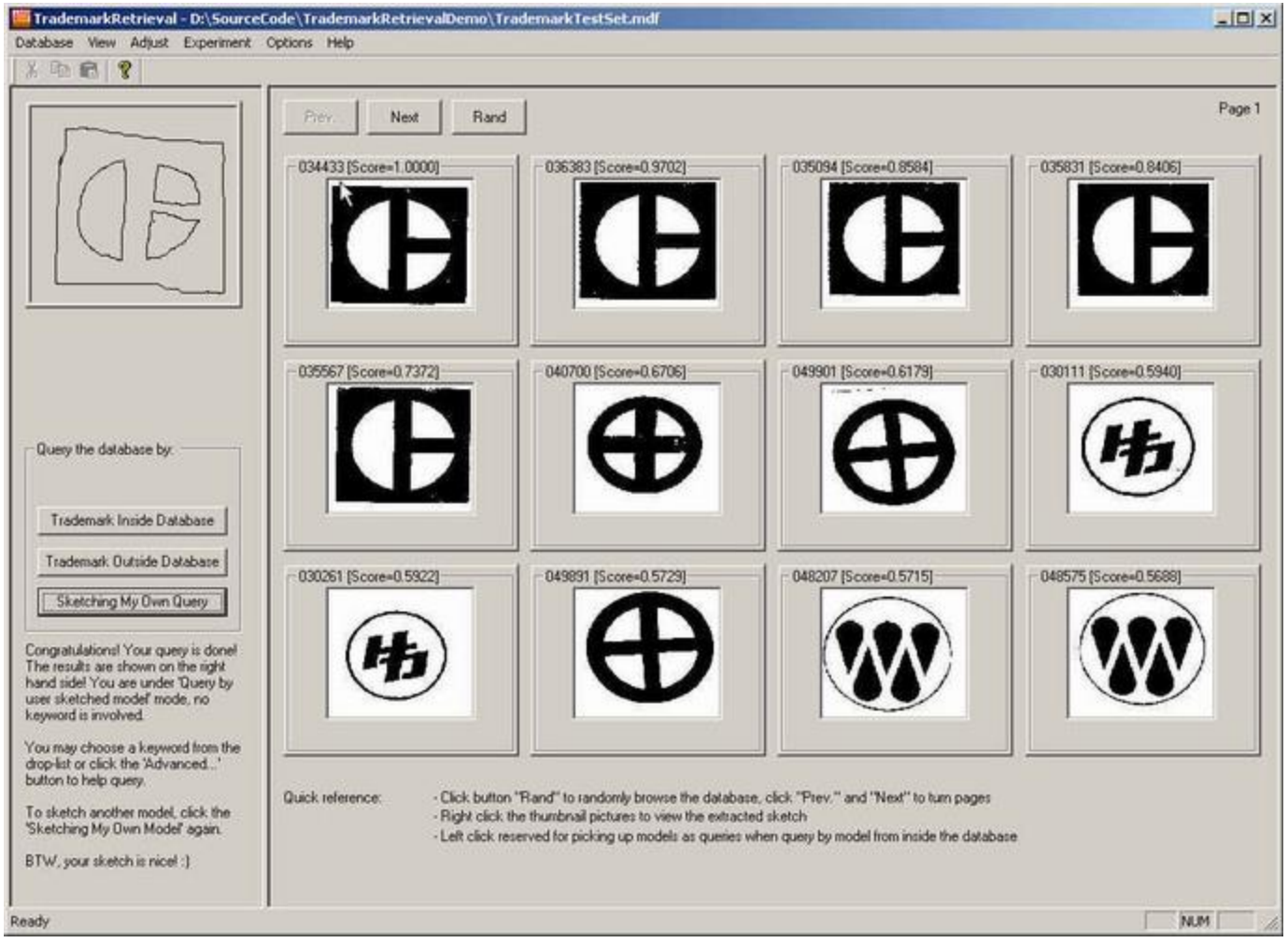


- Perimeter measurement;
- Area attribute;
- Roundness, or compactness

$$\gamma = \frac{(\text{perimeter})^2}{4\pi(\text{area})}$$

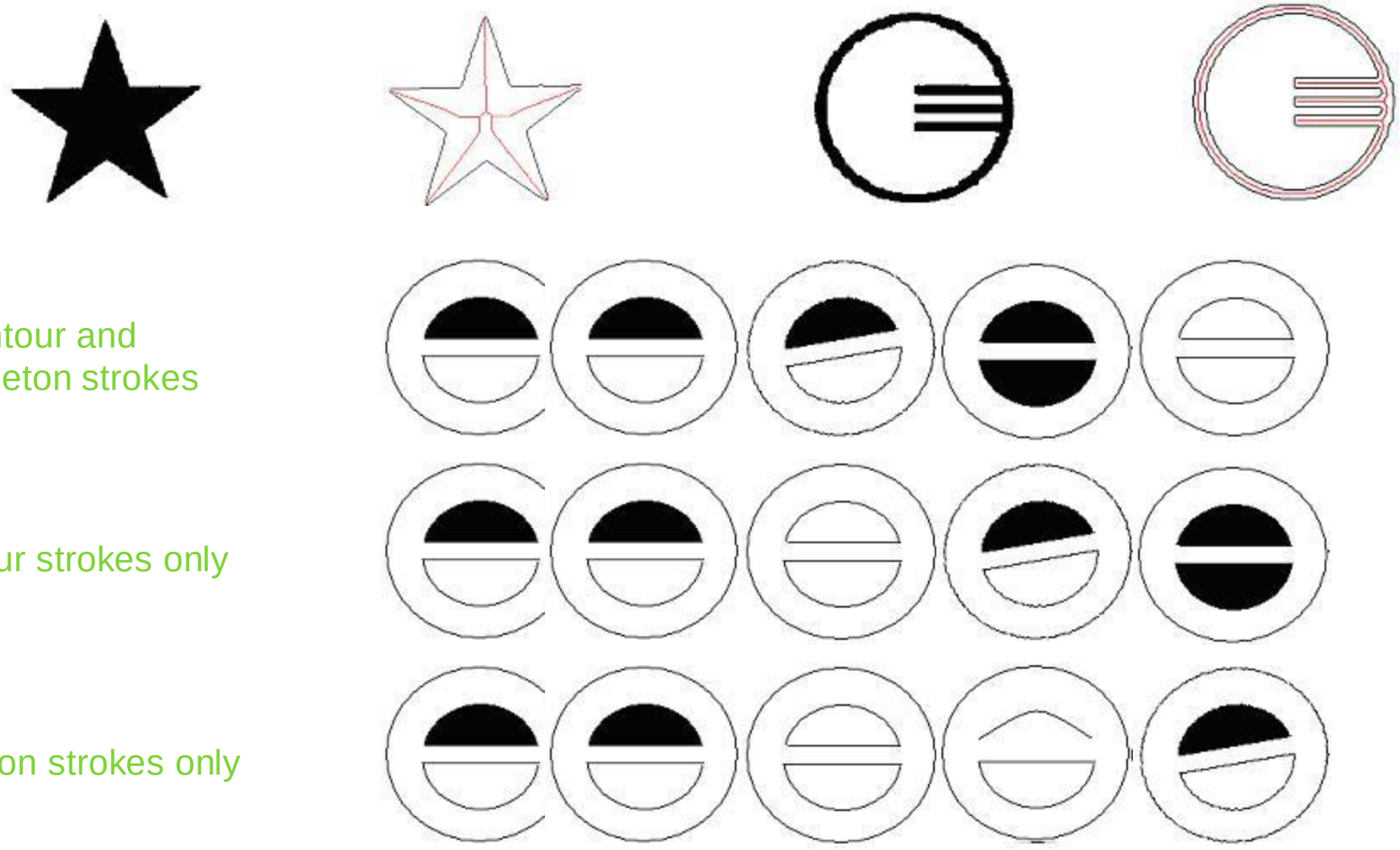
  - For a disc, the roundness parameter is minimum and equals 1.

Trademark Retrieval System



http://amp.ece.cmu.edu/projects/TrademarkRetrieval/

Trademark Retrieval System



Contour and skeleton strokes

Contour strokes only

Skeleton strokes only

VIIR System

DEMO

Need To Know

- Motivation of content-based multimedia retrieval
- Audio retrieval/classification/recognition
- Color features: color space, color histogram, color moments, CCV
- Texture features: characterizing and computation methods
- Shape features: representation methods

Reference Readings

- Multimedia information retrieval and management: technological fundamentals and applications, Springer, 2003.
  - Chapter 5, pp.95-108
  - Chapter 1, pp.1-16
  - Chapter 20, pp.438-446
  - Chapter 18, pp.385-393