图像的特征提取

数学与统计学院

2020年8月

Outlines

A Computer Vision System

Definition of Feature

Edge detection

Shape

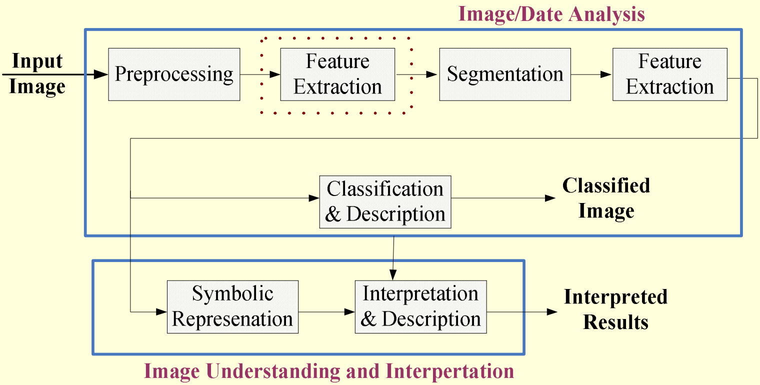


Figure 1: Abridge View of Computer Vision System

Image Analysis Techniques

Three broad categories:

- Feature Detection/Extraction
- Segmentation
- Classification

Computer Vision Applications

	Applications	Fields
1	Mail sorting, label reading, bank-check processing, text reading	Character Recognition
2	Tumor detection, blood cell count, chromosome analysis	Medical image Analysis
3	Parts identification on assembly lines, defect and fault inspection	Industrial automation
4	Multispectral image analysis, weather prediction, classification and monitoring of urban, agricultural and marine environments from satellite images	Remote Sensing

What is a Feature?

- In computer vision, a feature is a piece of information which is relevant for solving the computational task related to a certain application.
- It may be specific structures in the image such as points, edges or objects.
- It may be shapes defined in terms of curves or boundaries between different image regions, or properties of such a region
- The feature concept is very general and the choice of features in a particular computer vision system may be highly dependent on the specific problem at hand.

Image Features

- Spatial Features
  - Amplitude Features
  - Histogram Features
- Edges and Boundaries
- Shape Features
- Texture

Amplitude Features

- Gray level images:
  physical properties such as reflectivity, transmissivity, tristimulus values, etc.
  $f(x,y) = i(x,y) \cdot r(x,y)$
  where $i(x,y)$:illumination, $r(x,y)$:reflectance
- Medical images:
  the absorption characteristics of the body masses and enables discrimination bones from tissues.

$$\begin{array}{lll} Bone & \rightarrow High\ absorption & \rightarrow Higher\ pixel\ values \\ Water/Blood & \rightarrow Medium\ absorption & \rightarrow Moderate\ pixel\ values \\ Air & \rightarrow Almost\ no\ absorption & \rightarrow Low\ pixel\ values \end{array}$$

- Radar images:
  radar cross section

Color feature

Denote total number of pixels in the image as N, the number of pixels with gray level l as n_l. Assuming the gray scale is L, then average of the k_{th} channel color u_k, mean square deviation σ_k, entropy e_k can be calculated as

$$u_k = \frac{1}{N}\sum f_k(i,j) \tag{1}$$

$$\sigma_k = \sqrt{\frac{1}{N}\sum(f_k(i,j) - u_k)^2} \tag{2}$$

$$p_l = \frac{n_l}{N}, l = 1, 2, ..., L$$

$$e_k = -\sum_{l=1}^{L} p_l \log_2^{p_l} \tag{3}$$

Histogram

The Histogram of an image is defined as the array

$$h[l] = n_l \tag{4}$$

It can be easily verified that

$$N = \sum_{l=1}^{L} h[l] \tag{5}$$

where N is the total number of pixels in the image.
The Probability Density Function (pdf) of the gray levels can be expressed as

$$p[l] = \frac{h[l]}{N} \tag{6}$$

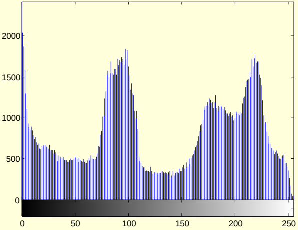
It is called normalized histogram.

The histogram provides a natural bridge between images and a probabilistic description.

Histogram–Cont'd

- Histogram is independent of the position of the pixel.
- The histogram is often displayed as a bar graph.
- The histogram is usually the only global image information available.
- It is used when finding optimal illumination conditions for capturing an image, gray scale transformations, and image segmentation to objects and background.
- One histogram may correspond to several images.

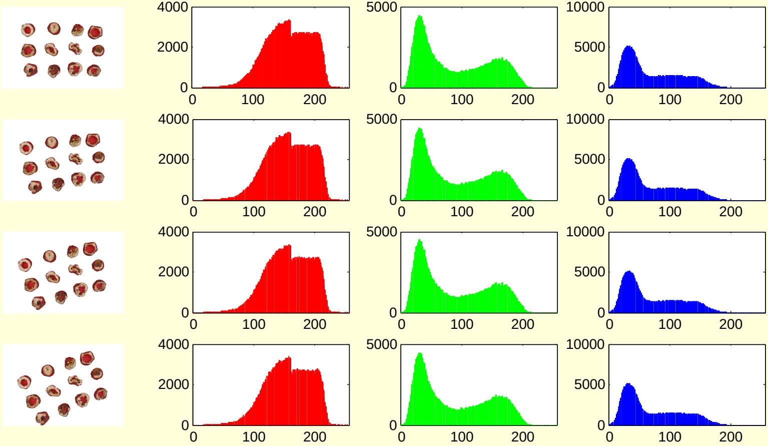
(a)



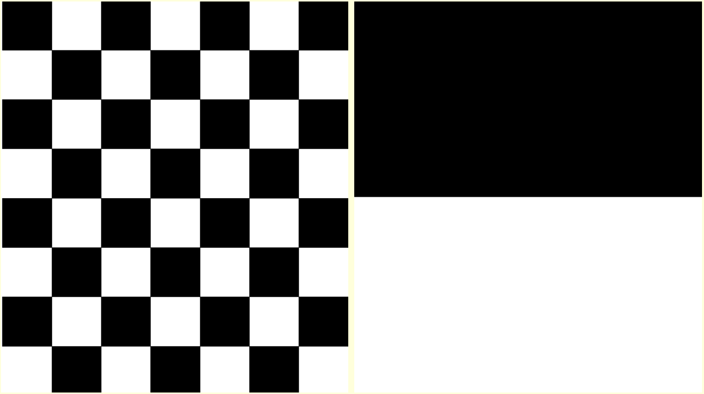
(b)

Merit of Histogram

A change of the object position on a constant background does not affect the histogram.



Limitation of Histogram–Cont'd



(a) (b)

Figure 3: Two Black & White images with the same histogram

Features of Histogram

If $f(x)$ is the pdf, the following moments can be used as features

Regular Moments	$M_k = \int_{-\infty}^{\infty} x^k f(x)dx$
Normalized Moments	$\mu_k = \int_{-\infty}^{\infty} (x-\bar{x})^k f(x)dx$
Central Moments	$\beta_k = (\mu_k)^{\frac{1}{k}}$

Specially,

- M_1:**Mean**
- M_2:**Average Energy**
- μ_2:**Variance**
- μ_3:**Skewness**

Geometric Moments

- The histogram reduces the description of an image to the specification of a 1-D function.
- The structural property of the images is not exploited.
- Better performance can be achieved by treating the image as a 2-D function.
- A 2-D image can be represented by various 2-D descriptors, such as 2-D moments and 2-D DFT.
- 2-D moments are very popular for pattern recognition.

2-D Geometric Moments

For a 2-D continuous function $f(x, y)$, the regular, central and central normalized moments of order are defined as:

$$M_{pq} = \int_{\infty}^{-\infty} \int_{\infty}^{-\infty} x^p y^q f(x, y) dxdy \qquad p, q = 0, 1, 2, \ldots$$

$$u_{pq} = \int_{\infty}^{-\infty} \int_{\infty}^{-\infty} (x - \overline{x})^p (y - \overline{y})^q f(x, y) dxdy \qquad p + q > 1$$

where $\overline{x} = M_{10}/M_{00}, \overline{y} = M_{01}/M_{00}$

$$\eta_{pq} = \frac{\mu_{pq}}{(\mu_{00})^{(p+q+2)/2}} \qquad p + q > 1$$

The normalized moments are used to nullify the effect of exponential growth of moment-magnitudes with increasing order.

2-D Invariant Moments

M. K. Hu proposed seven moments which are translation, scale and rotation invariant:

$$\begin{cases} I_1 = & \eta_{20} + \eta_{02} \\ I_2 = & (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\ I_3 = & (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\ I_4 = & (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\ I_5 = & (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ & +(3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\ I_6 = & (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\ & +4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\ I_7 = & (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ & -(\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \end{cases}$$

Edges characterize object boundaries and are useful for segmentation, registration, and identification of objects in a scene

- Gradient Operators
- Laplace Operators and Zero Crossings
- Canny Edge Detectors

Gradient Operators

$$\nabla f \equiv grad(f) = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

$$\frac{\partial f}{\partial x} \approx f(x+1, y) - f(x, y)$$

We will calculate:

$$\begin{array}{ll} \text{Gradient Magnitude} & M(x,y) = |\nabla f| = \sqrt{g_x^2 + g_y^2} \\ \text{Gradient Direction} & \alpha(x,y) = tan^{-1}\left(\frac{g_y}{g_x}\right) \end{array}$$

The Gradient magnitude can be thresholded to obtain an edge map.

Common Gradient Operators

$$
\begin{array}{ccc}
 & h_1 & h_2 \\
 & \text{for Vertical Edge} & \text{for Horizontal Edge} \\
\text{Prewitt} & \begin{bmatrix} -1 & 0 & 1 \\ -1 & \mathbf{0} & 1 \\ -1 & 0 & 1 \end{bmatrix} & \begin{bmatrix} -1 & -1 & -1 \\ 0 & \mathbf{0} & 0 \\ 1 & 1 & 1 \end{bmatrix} \\
 & & \\
\text{Sobel} & \begin{bmatrix} -1 & 0 & 1 \\ -2 & \mathbf{0} & 2 \\ -1 & 0 & 1 \end{bmatrix} & \begin{bmatrix} -1 & -2 & -1 \\ 0 & \mathbf{0} & 0 \\ 1 & 2 & 1 \end{bmatrix}
\end{array}
$$

Such h_1, h_2 are called convolution mask. Note bolded element indicates the location of the origin of the mask.

Edge Detection using Gradient Operator

(a) (b) (c)

Figure 4: (a) Original image (b) Gradient magnitude Map (c) Edge map using the threshold 36.5

Laplace Operators

- Gradient operators work best when the edge is sharp.
- For a wide transition region, it is more advantageous to apply second order derivatives, such as Laplacian operators.

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

- Because of the second order derivative, this gradient operator is more sensitive to noise.
- A better utilization of the Laplacian is to use its zero crossings to detect the edge locations.

Discrete Laplace Operators

$$
\underset{L_1}{\begin{bmatrix} 0 & -1 & 0 \\ -1 & \mathbf{4} & -1 \\ 0 & -1 & 0 \end{bmatrix}}
\quad
\underset{L_2}{\begin{bmatrix} -1 & -1 & -1 \\ -1 & \mathbf{8} & -1 \\ -1 & -1 & -1 \end{bmatrix}}
\quad
\underset{L_3}{\begin{bmatrix} 1 & -2 & 1 \\ -2 & \mathbf{4} & -2 \\ 1 & -2 & 1 \end{bmatrix}}
$$

(a) (b)

Figure 5: (a) Original image (b) Edge Map using zero-cross method

Canny Algorithm

- Filter the image to reduce noises
- Calculate intensity gradients of the image
- Apply non-maximum suppression to perform edge thinning and get rid of spurious response to edge detection
- Apply double threshold to determine strong & weak edges, and finalize the detection of edges by suppressing all the other edges that are weak and not connected to strong edges

(a) (b)

Figure 6: (a) Original image (b) Edge Map using the Canny method

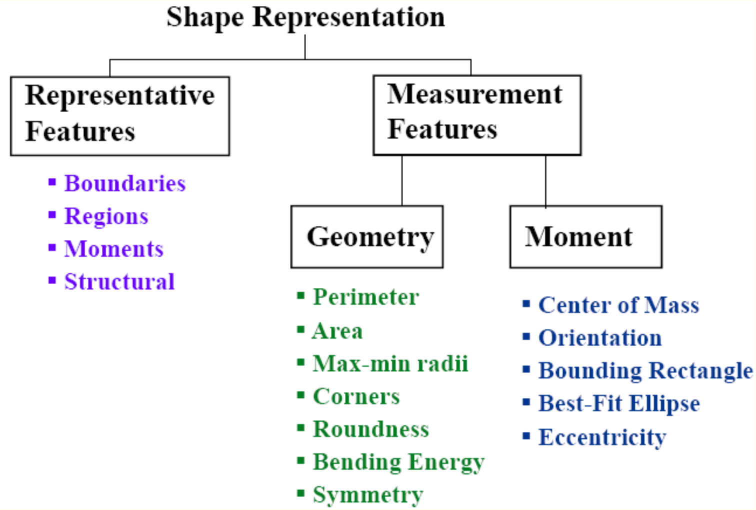
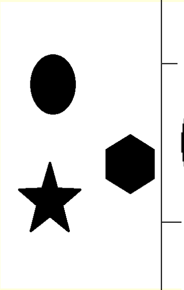
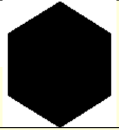


Figure 7: Measure Properties of Image Regions

Perimeter and Area

- Perimeter: The length of a closed contour of an object
- Area: The measurement of the surface occupied by an object

$$Area = \iint_{\mathbb{R}} dxdy$$

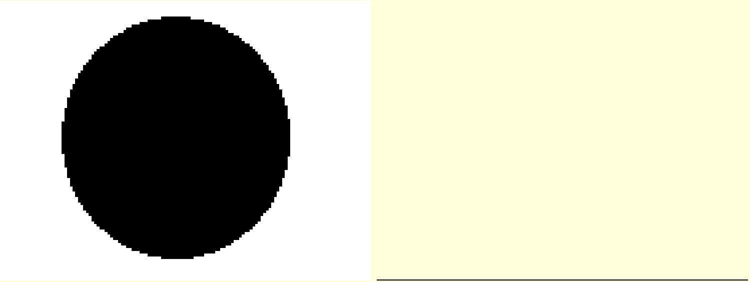
Original	Regions	Perimeter (pel)	Area (pel)
		505.75	486
		333.058	317
		343.7020	310

Roundness or Circularity

$$r = \frac{p^2}{4 * \pi * Area} \quad \text{p = Perimeter}$$

$$r = 1 \quad \text{for Circular Shape}$$

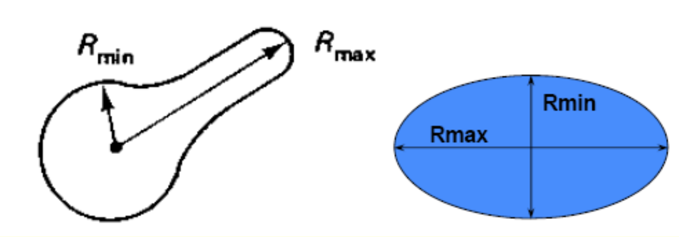
$$r = \infty \quad \text{for very thin object}$$



(a) (b)

Figure 8: (a) circle with $r = 1.12$ (b) thin object with $r = 212.42$

Shape Representation by Eccentricity



- R_{min}: Minimum distance to the boundary from the center of mass
- R_{max}: Maximum distance to the boundary from the center of mass
- $Eccentricity = \frac{R_{max}}{R_{min}}$

? Question: How to calculate the center of mass? How to calculate eccentricity?

This definition is slightly different from mathematical one, which is employed in Matlab.

Curvature Functions for Corner Detection

Corners are locations on the boundary where the curvature $\kappa(t)$ becomes unbounded.

$$|\kappa(t)|^2 \equiv \left(\frac{d^2y}{dt^2}\right)^2 + \left(\frac{d^2x}{dt^2}\right)^2$$

? **Recall its mathematical form.**

A corner is detected when $|\kappa(t)| > T$.

Moment Based Features

- Center of Mass
- Orientation
- Boundary Rectangle
- Best Fit Ellipse
- Eccentricity

Moment Based Features

The two-dimensional moment for a $(N \times M)$ discretized image, $g(x,y)$, is

$$m_{pq} = \sum_{y=0}^{M-1} \sum_{x=0}^{N-1} x^p y^q g(x,y)$$

Properties of Low-Order Moments:

- Zero order moments: m_{00} is the total mass of an image. Specially, $g(x,y) = 1$ in binary image, the zeroth moment m_{00} represents the total object area.
- First order moments: the coordinates of the center of mass are

$$\bar{x} = \frac{m_{10}}{m_{00}}, \bar{y} = \frac{m_{01}}{m_{00}}$$

Central moments are designated by u_{pq}:

$$u_{pq} = \sum_{y=0}^{M-1} \sum_{x=0}^{N-1} (x-\bar{x})^p (y-\bar{y})^q g(x,y)$$

Second Order Moments

- Principal Axes: the orientation of the principal axes, θ, is given by

$$\theta = \frac{1}{2}\tan^{-1}\left(\frac{2\mu_{11}}{\mu_{20}-\mu_{02}}\right), \frac{-\pi}{4} \le \theta \le \frac{\pi}{4}$$

The angle of major principal axis, ϕ, may be determined by the angle θ.

Table 1: Orientation of the Major Principal Axis.

μ_{11}	$\mu_{20}-\mu_{02}$	θ	ϕ
0	-	0	$\pi/2$
+	-	$0 > \theta > -\pi/4$	$\pi/2+\theta$
+	0	0	$\pi/4$
+	+	$\pi/4 > \theta > 0$	θ
0	0	0	0
-	+	$0 > \theta > -\pi/4$	θ
-	0	0	$-\pi/4$
-	-	$\pi/4 > \theta > 0$	$-\pi/2+\theta$

Image Ellipse

The image ellipse is a constant intensity elliptical disk with the same mass and second order moments as the original image. If the image ellipse is defined with semi-major axis, α, along the x axis and semi-minor axis, β, along the y axis, then α and β may be determined from the second order moments using

$$\alpha = \left(\frac{2\left[\mu_{20}+\mu_{02}+\sqrt{(\mu_{20}-\mu_{02})^2+4\mu_{11}^2}\right]}{\mu_{00}} \right)^{1/2}$$

$$\beta = \left(\frac{2\left[\mu_{20}+\mu_{02}-\sqrt{(\mu_{20}-\mu_{02})^2+4\mu_{11}^2}\right]}{\mu_{00}} \right)^{1/2}$$

The intensity of the image ellipse is then given by

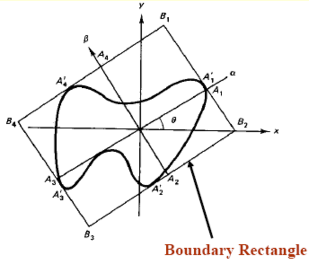
$$I = \frac{\mu_{00}}{\pi\alpha\beta}$$

Bounding rectangle

- Up-right bounding rectangle for the specified point set.
- Rotated rectangle of the minimum area enclosing the specified point set. The bounding rectangle is the smallest rectangle enclosing the object that is also aligned with its orientation. Once the orientation θ is known, we use the transformation

$$\alpha = x\cos\theta + y\sin\theta$$
$$\beta = -xsin\theta + y\cos\theta$$

on the boundary points and search for α_{min}, α_{max}, β_{min}, β_{max}, which are corresponding points A'_3, A'_1, A'_2, A'_4 in the bounding rectangle diagram.



Bounding rectangle–Cont’d

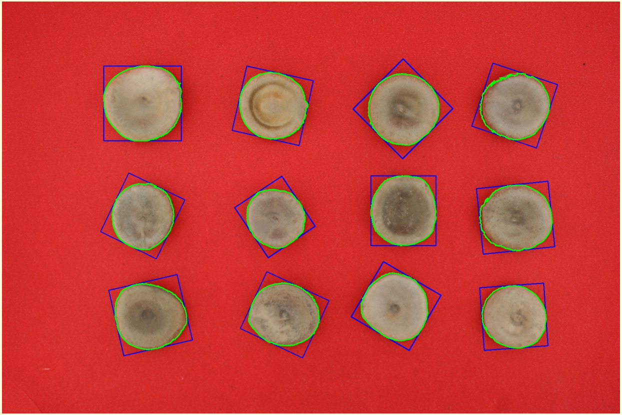


Figure 9: Bounding Rectangle highlighted in blue

Local Binary Pattern

- Texture is an image attribute which describes properties such as smoothness, coarseness, regularity, etc. It shows the organizational structure of the surface and its sequence. Unifying local binary pattern (LBP) and gray level co-occurrence matrix will be discussed.
- Local binary pattern is a convincing texture description which is widely used in many areas of image processing such as face recognition and defect detection, etc.
- Suppose current pixel as the center of a neighbor, and compare gray value of the pixels around within a certain neighbor radius R. The local binary pattern LBP is defined as

$$\mathrm{LBP}_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p, s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

  where P is the total number of points in the neighbor, g_c and g_p refers to the gray value of pixels on the centre and boundary respectively.

Unified Binary Pattern

Ojala proposed an unified binary pattern $\mathrm{LBP}_{P,R}^{\mathrm{riu2}}$:

$$\mathrm{LBP}_{P,R}^{\mathrm{riu2}} = \begin{cases} \sum_{p=0}^{P-1} s(g_p - g_c), & \textit{if } u(\mathrm{LBP}_{P,R}) \leq 2 \\ P+1, & \textit{else} \end{cases}$$

$$u(\mathrm{LBP}_{P,R}) = |s(g_{p-1} - g_c) - s(g_0 - g_c)| + \sum_{p=1}^{P-1} |s(g_p - g_c) - s(g_{p-1} - g_c)|$$

Actually $u(.)$ calculates the number of transition of binary presentation in LBP.

Gray Level Co-occurrence Matrix

- Gray level co-occurrence matrix(GLCM) is a common approach to describe texture by studying the related spatial features of gray level.
- Suppose points (x, y) and $(x + a, y + b)$ in an image, if their gray values are i and j respectively, a gray value pair is denoted by (i, j). Taking different values of (a, b), along a certain direction θ and at a certain interval $d = \sqrt{a^2 + b^2}$, frequency of an (i, j) can be counted, with the notation $p(i, j, d, \theta)$. All such $p(i, j, d, \theta)$ construct a gray level co-occurrence matrix, denoted by $\mathbf{P}(i, j, d, \theta)$.
- Normally the θ takes 0^o, 45^o, 90^o and 135^o.

Feature Based on GLCM

Two second moments W_1, contrast W_2, relativity W_3 and entropy W_4 can be coined to serve as texture feature.

Angular second moment:

$$W_1 = (\sum_{k=0}^{3}\sum_{i=1}^{g}\sum_{j=1}^{g}\mathbf{P}^2(i,j,d,k\frac{\pi}{4}))/4$$

where g is the gray scale. Contrast:

$$W_2 = (\sum_{k=0}^{3}\sum_{i=1}^{g}\sum_{j=1}^{g}[(i-j)^2\mathbf{P}^2(i,j,d,k\frac{\pi}{4})])/4$$

Correlation:

$$W_3 = (\sum_{k=0}^{3}\sum_{i=1}^{g}\sum_{j=1}^{g}\frac{ij\mathbf{P}(i,j,d,k\frac{\pi}{4}) - u_1(k)u_2(k)}{d_1^2(k)d_2^2(k)})/4$$

where

$$u_1(k) = \sum_{i=1}^{g} i \sum_{j=1}^{g}\mathbf{P}(i,j,d,k\frac{\pi}{4}), u_2(k) = \sum_{i=1}^{g} j \sum_{j=1}^{g}\mathbf{P}(i,j,d,k\frac{\pi}{4})$$

Feature Based on GLCM–Cont'd

$$d_1(k) = \sum_{i=1}^{g}(i - u_1(k))^2 \sum_{j=1}^{g} \mathbf{P}(i,j,d,k\frac{\pi}{4}),$$

$$d_2(k) = \sum_{i=1}^{g}\sum_{j=1}^{g}(j - u_2(k))^2 \mathbf{P}(i,j,d,k\frac{\pi}{4})$$

Entropy:

$$W_4 = -(\sum_{k=0}^{3}\sum_{i=1}^{g}\sum_{j=1}^{g} P(i,j,d,k\frac{\pi}{4})\log(P(i,j,d,k\frac{\pi}{4})))/4$$

References

1. M. K. Mandal. lecture notes. Winter 2016.
2. Gonzalez, Rafael C., and Richard E. Woods. "Image processing." Digital image processing 2 (2007).
3. Sonka, Milan, Vaclav Hlavac, and Roger Boyle. Image processing, analysis, and machine vision. Cengage Learning, 2014.
4. Prokop, Richard J., and Anthony P. Reeves. "A survey of moment-based techniques for unoccluded object representation and recognition." CVGIP: Graphical Models and Image Processing 54.5 (1992): 438-460.