Statistical Texture Measures Computed from Gray Level Coocurrence Matrices

Fritz Albregtsen
Image Processing Laboratory
Department of Informatics
University of Oslo

November 5, 2008

Abstract

The purpose of the present text is to present the theory and techniques behind the Gray Level Coocurrence Matrix (GLCM) method, and the state-of-the-art of the field, as applied to two dimensional images. It does not present a survey of practical results.

1 Gray Level Coocurrence Matrices

In statistical texture analysis, texture features are computed from the statistical distribution of observed combinations of intensities at specified positions relative to each other in the image. According to the number of intensity points (pixels) in each combination, statistics are classified into first-order, second-order and higher-order statistics.

The Gray Level Coocurrence Matrix (GLCM) method is a way of extracting second order statistical texture features. The approach has been used in a number of applications, e.g. [5],[6],[14],[5],[7],[12],[2],[8],[10],[1].

A GLCM is a matrix where the number of rows and colums is equal to the number of gray levels, G, in the image. The matrix element $P(i, j \mid \Delta x, \Delta y)$ is the relative frequency with which two pixels, separated by a pixel distance $(\Delta x, \Delta y)$, occur within a given neighborhood, one with intensity i and the other with intensity j. One may also say that the matrix element $P(i, j \mid d, \theta)$ contains the second order statistical probability values for changes between gray levels i and j at a particular displacement distance d and at a particular angle (θ).

Given an $M \times N$ neighborhood of an input image containing G gray levels from 0 to $G-1$, let $f(m,n)$ be the intensity at sample m, line n of the neighborhood. Then

$$P(i,j \mid \Delta x, \Delta y) = WQ(i,j \mid \Delta x, \Delta y) \tag{1}$$

where

$$W = \frac{1}{(M-\Delta x)(N-\Delta y)}$$

$$Q(i,j \mid \Delta x, \Delta y) = \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} A$$

and

$$A = \begin{cases} 1 & \text{if } f(m,n) = i \text{ and } f(m+\Delta x, n+\Delta y) = j \\ 0 & \text{elsewhere} \end{cases}$$

A small (5×5) sub-image with 4 gray levels and its corresponding gray level coocrurrence matrix $P(i,j \mid \Delta x = 1, \Delta y = 0)$ is illustrated below.

```
IMAGE                      P(i,j;1,0)

0 1 1 2 3                    j=0   1    2    3
0 0 2 3 3               i= 0  1/20 2/20 1/20  0
0 1 2 2 3                  1   0   1/20 3/20  0
1 2 3 2 2                  2   0    0   3/20 5/20
2 2 3 3 2                  3   0    0   2/20 2/20
```

Using a large number of intensity levels G implies storing a lot of temporary data, i.e. a $G \times G$ matrix for each combination of $(\Delta x, \Delta y)$ or (d, θ). One sometimes has the paradoxical situation that the matrices from which the texture features are extracted are more voluminous than the original images from which they are derived. It is also clear that because of their large dimensionality, the GLCM's are very sensitive to the size of the texture samples on which they are estimated. Thus, the number of gray levels is often reduced.

Even visually, quantization into 16 gray levels is often sufficient for discrimination or degmentation of textures. Using few levels is equivalent to viewing the image on a coarse scale, whereas more levels give an image with more detail. However, the performance of a given GLCM-based feature, as well as the ranking of the features, may depend on the number of gray levels used.

Because a $G \times G$ matrix (or histogram array) must be accumulated for each sub-image/window and for each separation parameter set (d, θ), it is usually computationally necessary to restrict the (d, θ)-values to be tested to a limited number of values. Figure 1 below illustrates the geometrical relationships of GLCM measurements made for four distances d ($d = max\{\mid \Delta x \mid, \mid \Delta y \mid\}$) and angles of $\theta = 0, \pi/4, \pi/2$ and $3\pi/4$ radians under the assumption of angular symmetry.

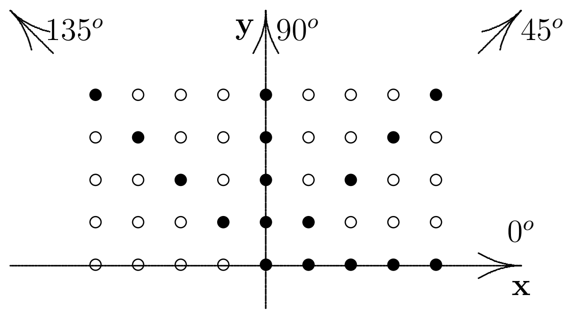

Figure 1: *Geometry for measurement of gray level coocurrence matrix for 4 distances d and 4 angles θ.*

In order to obtain a statistically reliable estimate of the joint probability distribution, the matrix must contain a reasonably large average occupancy level. This can be achieved either by restricting the number of gray value quantization levels or by using a relatively large window. The former approach results in a loss of texture description accuracy in the analysis of low amplitude textures, while the latter causes uncertainty and error if the texture changes over the large window. A typical compromize is to use 16 gray levels and a window of about 30 to 50 pixels on each side.

Simple relationships exist among certain pairs of the estimated probability distributions $P(d, \theta)$. Let $P^t(d, \theta)$ denote the transpose of the matrix $P(d, \theta)$. Then

$$P(d, 0^0) = P^t(d, 180^0)$$

$$P(d, 45^0) = P^t(d, 225^0)$$
$$P(d, 90^0) = P^t(d, 270^0)$$
$$P(d, 135^0) = P^t(d, 315^0)$$

Thus, the knowledge of $P(d, 180^0)$, $P(d, 225^0)$, $P(d, 270^0)$, and $P(d, 315^0)$ adds nothing to the specification of the texture.

For a given distance d we usually have four angular gray level coocurrence matrices. Haralick et al. 1973 [5] and Conners et al. 1984 [2] have proposed a number of *scalar* texture measures $T(d, \theta)$ that may be extracted from these matrices (see below). If one wants to avoid dependency of direction one may calculate an average (isotropic) matrix out of four matrices, $\theta = 0^0$, 45^0, 90^0, 135^0. Haralick et al. 1973 [5] have suggested to use the angular mean, $M_T(d)$, and range, $R_T(d)$, of each of the proposed textural measures, T, as a set of features used as input to a classifier:

$$M_T(d) = \frac{1}{N_\theta} \sum_\theta T(d, \theta) \tag{2}$$

$$R_T(d) = \max_\theta \left[T(d, \theta)\right] - \min_\theta \left[T(d, \theta)\right] \tag{3}$$

where the summation is over the angular measurements, and N_θ represents the number of such measurements (here 4). Similarly, an angular independent texture variance may be defined as

$$V_T^2(d) = \frac{1}{N_\theta} \sum_\theta \left[T(d, \theta) - M_T(d)\right]^2 \tag{4}$$

Within the large number of texture features available, some of the features are strongly correlated with each other. A feature selection procedure may be applied in order to select a subset or a linear combination of the features available, either using a set of training image regions to establish the set of features giving the smallest classification error, or using some functional feature space distance metric such that a large feature space distance implies a small classification error. Tou et al. 1977 [**?**] proposed to use the Karhunen-Loeve expansion to extract optimal properties from the full property set. Gotlieb and Kreyszig 1990 [4] found that groups of four features were optimal.

Conners and Harlow 1980 [1] concluded that the coocurrence matrix approach cannot innately discriminate between all visual texture pairs if only one intersample spacing distance is utilized. This suggests that a number of intersample spacing distances should be used for more accurate classification. An alternative is to use only the best feature(s), and evaluate this very limited set of features for all intersample distances up to a practical limit (e.g. $d = 4$), see Wu and Chen 1992 [15].

2 Texture Features from GLCM

A number of texture features may be extracted from the GLCM (see Haralick et al. 1973 [5], Conners et al. 1984 [2]). We use the following notation:
G is the number of gray levels used.
μ is the mean value of P.
μ_x, μ_y, σ_x and σ_y are the means and standard deviations of P_x and P_y. $P_x(i)$ is

the ith entry in the marginal-probability matrix obtained by summing the rows of $P(i,j)$:

$$P_x(i) = \sum_{j=0}^{G-1} P(i,j)$$

$$P_y(j) = \sum_{i=0}^{G-1} P(i,j)$$

$$\mu_x = \sum_{i=0}^{G-1} i \sum_{j=0}^{G-1} P(i,j) = \sum_{i=0}^{G-1} iP_x(i)$$

$$\mu_y = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} jP(i,j) = \sum_{j=0}^{G-1} jP_y(j)$$

$$\sigma_x^2 = \sum_{i=0}^{G-1} (i-\mu_x)^2 \sum_{j=0}^{G-1} P(i,j) = \sum_{i=0}^{G-1} (P_x(i) - \mu_x(i))^2$$

$$\sigma_y^2 = \sum_{j=0}^{G-1} (j-\mu_y)^2 \sum_{i=0}^{G-1} P(i,j) = \sum_{j=0}^{G-1} (P_y(j) - \mu_y(j))^2$$

and

$$P_{x+y}(k) = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i,j) \quad i+j=k \tag{5}$$

for $k = 0, 1, ..., 2(G-1)$.

$$P_{x-y}(k) = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i,j) \quad \mid i-j \mid = k \tag{6}$$

for $k = 0, 1, ..., G-1$.

The following features are used:

- Homogeneity, Angular Second Moment (ASM) :

$$ASM = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \{P(i,j)\}^2 \tag{7}$$

ASM is a measure of homogeneity of an image. A homogeneous scene will contain only a few gray levels, giving a GLCM with only a few but relatively high values of $P(i,j)$. Thus, the sum of squares will be high.

- Contrast :

$$CONTRAST = \sum_{n=0}^{G-1} n^2 \{ \sum_{i=1}^{G} \sum_{j=1}^{G} P(i,j) \}, \quad | i - j | = n \tag{8}$$

This measure of contrast or local intensity variation will favour contributions from $P(i,j)$ away from the diagonal, i.e. $i \neq j$.

- Local Homogeneity, Inverse Difference Moment (IDM) :

$$IDM = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{1}{1 + (i-j)^2} P(i,j) \tag{9}$$

IDM is also influenced by the homogeneity of the image. Because of the weighting factor $(1+(i-j)^2)^{-1}$ IDM will get small contributions from inhomogeneous areas $(i \neq j)$. The result is a low IDM value for inhomogeneous images, and a relatively higher value for homogeneous images.

- Entropy :

$$ENTROPY = - \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i,j) \times log(P(i,j)) \tag{10}$$

Inhomogeneous scenes have low first order entropy, while a homogeneous scene has a high entropy.

- Correlation :

$$CORRELATION = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{\{i \times j\} \times P(i,j) - \{\mu_x \times \mu_y\}}{\sigma_x \times \sigma_y} \tag{11}$$

Correlation is a measure of gray level linear dependence between the pixels at the specified positions relative to each other.

- Sum of Squares, Variance :

$$VARIANCE = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} (i - \mu)^2 P(i,j) \tag{12}$$

This feature puts relatively high weights on the elements that differ from the average value of $P(i,j)$.

- Sum Average :

$$AVER = \sum_{i=0}^{2G-2} i P_{x+y}(i) \tag{13}$$

- Sum Entropy :

$$SENT = -\sum_{i=0}^{2G-2} P_{x+y}(i) log(P_{x+y}(i)) \qquad (14)$$

- Difference Entropy :

$$DENT = -\sum_{i=0}^{G-1} P_{x+y}(i) log(P_{x+y}(i)) \qquad (15)$$

- Inertia :

$$INERTIA = \sum_{i=0}^{G-1}\sum_{j=0}^{G-1} \{i-j\}^2 \times P(i,j) \qquad (16)$$

- Cluster Shade :

$$SHADE = \sum_{i=0}^{G-1}\sum_{j=0}^{G-1} \{i+j-\mu_x-\mu_y\}^3 \times P(i,j) \qquad (17)$$

- Cluster Prominence :

$$PROM = \sum_{i=0}^{G-1}\sum_{j=0}^{G-1} \{i+j-\mu_x-\mu_y\}^4 \times P(i,j) \qquad (18)$$

3 Features from sum and difference histograms

Unser [13] demonstrated that the sum and difference define the principal axes of the second-order probability function of a stationary process. Therefore, he suggested that the usual GLCM's used for texture description be replaced by their sum and difference histograms which can be estimated directly from the image. Some of the GLCM-based texture features can easily be obtained from such sum and difference histograms. Cluster Shade and Cluster Prominence can be modified to a sum histogram problem, while Inertia and Local Homogeneity can be obtained from difference histograms.

Given an $M \times N$ neighborhood D of an input image containing G gray levels from 0 to $G-1$, let $f(m,n)$ be the intensity at sample m, line n of the neighborhood. Then the normalized sum and difference histograms are given by

$$H_s(i \mid \Delta x, \Delta y) = WCard\{(m,n) \in D, s_{\Delta x, \Delta y}(m,n) = i\} \qquad (19)$$

$$H_d(j \mid \Delta x, \Delta y) = WCard\{(m,n) \in D, d_{\Delta x, \Delta y}(m,n) = j\} \qquad (20)$$

where

$$W = \frac{1}{(M - \Delta x)(N - \Delta y)}$$

$$s_{\Delta x,\Delta y}(m,n) = f(m,n) + f(m + \Delta x, n + \Delta y)$$

$$d_{\Delta x,\Delta y}(m,n) = f(m,n) - f(m + \Delta x, n + \Delta y)$$

The number of possible values in the histograms is $2G - 1$.

3.1 Cluster Shade

$$SHADE = \sum_{i=0}^{2G-2} (i - 2\mu)^3 H_s(i \mid \Delta x, \Delta y) \tag{21}$$

where

$$\mu = \frac{1}{2} \sum_{i=0}^{2G-2} i H_s(i \mid \Delta x, \Delta y)$$

3.2 Cluster Prominence

$$PROMINENCE = \sum_{i=0}^{2G-2} (i - 2\mu)^4 H_s(i \mid \Delta x, \Delta y) \tag{22}$$

3.3 Contrast

$$CONTRAST = \sum_{j=0}^{2G-2} j^2 H_d(j \mid \Delta x, \Delta y) \tag{23}$$

3.4 Homogeneity

$$HOMOGENEITY = \sum_{j=0}^{2G-2} \frac{1}{1 + j^2} H_d(j \mid \Delta x, \Delta y) \tag{24}$$

3.5 Correlation

$$CORRELATION = \frac{1}{2} \left(\sum_{i=0}^{2G-2} (i - 2\mu)^2 H_s(i \mid \Delta x, \Delta y) - \sum_{j=0}^{2G-2} j^2 H_d(j \mid \Delta x, \Delta y) \right) \tag{25}$$

4 Features from sum and difference images

For some of the GLCM texture features, no operations are to be performed on the probability distribution $P(i, j \mid \Delta x, \Delta y)$ that require the information to be in matrix or histogram form (see Peckinpaugh 1991 [9]). Using the Cluster Shade equations as an example, one may demonstrate that it is possible to reduce the number of addition operations greatly by expressing the texture features in terms of the image pixel values contained in a given neighborhood. These methods can be partly applied to the other GLCM-based texture measures listed above.

4.1 Cluster Shade

In the cluster shade equations, it is possible to reduce the number of operations by expressing this texture feature in terms of the image pixel values contained in a given $M \times N$ neighborhood containing G gray levels from 0 to $G-1$, where $f(m,n)$ is the intensity at sample m, line n of the neighborhood.

$$SHADE = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \{i + j - \mu_i - \mu_j\}^3 \times P(i, j \mid \Delta x, \Delta y) \tag{26}$$

where

$$\mu_i = \sum_{i=0}^{G-1} i \sum_{j=0}^{G-1} P(i, j \mid \Delta x, \Delta y)$$

$$\mu_j = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} j \times P(i, j \mid \Delta x, \Delta y)$$

Now we may write

$$\mu_i = W \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} f(m, n) \tag{27}$$

$$\mu_j = W \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} f(m + \Delta x, n + \Delta y) \tag{28}$$

And the cluster shade equation may be rewritten in terms of the neighborhood intensities and the μ_i and μ_j terms above:

$$SHADE = W \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} (f(m, n) + f(m + \Delta x, n + \Delta y) - \mu_{i+j})^3 \tag{29}$$

where

$$\mu_{i+j} = W \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} (f(m,n) + f(m+\Delta x, n+\Delta y))$$

Now let $g(m,n)$ be the intensity of the pixel at sample m, line n of a new "$i+j$" - image:

$$g(m,n) = f(m,n) + f(m+\Delta x, n+\Delta y) \tag{30}$$

The equations above can be rewritten in terms of this new image,

$$SHADE = W \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} (g(m,n) - \mu(g)_{i+j})^3 \tag{31}$$

where

$$\mu(g)_{i+j} = W \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} g(m,n)$$

To sum up: A new "$i+j$" - image is created, having a range of integer intensities from 0 to $2(G-1)$. No GLCM is accumulated. The value of $\mu(g)_{i+j}$ is computed and stored for the first neighborhood of the image, and is subsequently updated as the neighborhood is moved one pixel. For all $i+j$ image values of the neighborhood, the computed $\mu(g)_{i+j}$ is subtracted. This result is then cubed and added to the cluster shade sum value. Lastly, this sum value is weighted for the neighborhood size.

4.2 Cluster Prominence

Cluster Prominence is the easiest to relate to Cluster Shade. The original equation

$$PROM = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \{i+j-\mu_i-\mu_j\}^4 \times P(i,j \mid d) \tag{32}$$

where

$$\mu_i = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i,j \mid d), \qquad \mu_j = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} j \times P(i,j \mid d)$$

is rewritten in terms of the g - image:

$$PROM = W \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} (g(m,n) - \mu(g)_{i+j})^4 \tag{33}$$

where

$$\mu(g)_{i+j} = W \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} g(m,n)$$

$$g(m,n) = f(m,n) + f(m+\Delta x, n+\Delta y)$$

4.3 Inertia

$$INERTIA = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \{i - j\}^2 \times P(i, j \mid \Delta x, \Delta y) \tag{34}$$

is rewritten in terms of an $(i - j)^2$ - image:

$$INERTIA = W \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} g(m, n) \tag{35}$$

where

$$g(m, n) = (f(m, n) - f(m + \Delta x, n + \Delta y))^2$$

4.4 Local Homogeneity

For Local Homogeneity, or Inverse Difference Moment (IDM), the $g(m, n)$ values are more complicated to compute and must be real values. The original equation

$$IDM = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{1}{1 + (i - j)^2} P(i, j \mid \Delta x, \Delta y) \tag{36}$$

is rewritten as

$$IDM = W \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} g(m, n) \tag{37}$$

where

$$g(m, n) = \frac{W}{1 + (f(m, n) - f(m + \Delta x, n + \Delta y))^2}$$

4.5 Correlation

The original equation

$$CORR = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{\{i \times j\} \times P(i, j) - \mu_i \times \mu_j}{\sigma_i \times \sigma_j} \tag{38}$$

where

$$\mu_i = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i, j \mid \Delta x, \Delta y)$$

$$\mu_j = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} j \times P(i, j \mid \Delta x, \Delta y)$$

$$\sigma_i^2 = \sum_{i=0}^{G-1} (i - \mu_i)^2 \sum_{j=0}^{G-1} P(i, j \mid \Delta x, \Delta y)$$

$$\sigma_j^2 = \sum_{j=0}^{G-1} (j - \mu_j)^2 \sum_{i=0}^{G-1} P(i, j \mid \Delta x, \Delta y)$$

is rewritten as:

$$CORR = \frac{W}{\sigma_i \sigma j} \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} (f(m,n) - \mu_i)(f(m + \Delta x, n + \Delta y) - \mu_j) \tag{39}$$

where

$$\mu_i = W \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} f(m,n)$$

$$\mu_j = W \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} f(m + \Delta x, n + \Delta y)$$

$$\sigma_i^2 = R \left[T \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} f(m,n)^2 - \left(\sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} f(m,n) \right)^2 \right]$$

$$\sigma_j^2 = R \left[T \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} f(m + \Delta x, n + \Delta y)^2 - \left(\sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} f(m + \Delta x, n + \Delta y) \right)^2 \right]$$

$$T = (M - \Delta x)(N - \Delta y)$$

$$R = W \frac{1}{T(T-1)}$$

References

[1] R.W. Conners and C.A. Harlow,
"A theoretical comparison of texture algorithms",
IEEE Trans. on Pattern Analysis and Machine Intell., Vol. PAMI-2, pp. 204-222, 1980.

[2] R.W. Conners, M.M. Trivedi, and C.A. Harlow,
"Segmentation of a High-Resolution Urban Scene Using Texture Operators",
Computer Vision, Graphics, and Image Processing, Vol. 25, pp. 273-310, 1984.

[3] J.M.H. du Buf, M. Kardan and M. Spann,
"Texture Feature Performance of Image Segmentation",
Pattern Recognition, Vol. 23, pp. 291-309, 1990.

[4] JC.C. Gotlieb and H.E. Kreyszig,
"Texture Descriptors based on Co-ocurrence Matrices",
Computer Vision, Graphics, and Image Processing, Vol. 51, pp. 70-86, 1990.

[5] R.M. Haralick, K. Shanmugam, and I. Dinstein,
"Textural Features for Image Classification",
IEEE Trans. on Systems, Man and Cybernetics, Vol. SMC-3, pp. 610-621, 1973.

[6] R.M. Haralick and K. Shanmugam,
"Computer Classification of Reservoir Sandstones",
IEEE Trans. on Geo. Eng., Vol. GE-11, pp. 171-177, 1973.

[7] D.C. He, L.Wang, and J. Juibert,
"Texture Feature Extraction",
Pattern Recognition Letters, Vol. 6, pp. 269-273, 1987.

[8] M. Iizulca,
"Quantitative evaluation of similar images with quasi-gray levels",
Computer Vision, Graphics, and Image Processing, Vol. 38, pp. 342-360, 1987.

[9] S. Peckinpaugh,
"An Improved Method for Computing Gray-Level Coocurrence Matrix Based Texture Measures",
Computer Vision, Graphics, and Image Processing; Graphical Models and Image Processing, Vol. 53, pp. 574-580, 1991.

[10] L.H. Siew, R.H. Hodgson, and E.J. Wood,
"Texture Measures for Carpet Wear Asessment",
IEEE Trans. on Pattern Analysis and Machine Intell., Vol. PAMI-10, pp. 92-105, 1988.

[11] L.H. Siew, R.H. Hodgson, and E.J. Wood,
"Texture Measures for Carpet Wear Asessment",
IEEE Trans. on Pattern Analysis and Machine Intell., Vol. PAMI-10, pp. 92-105, 1988.

[12] M.M. Trivedi, R.M. Haralick, R.W. Conners, and S. Goh,
"Object Detection based on Gray Level Coocurrence",
Computer Vision, Graphics, and Image Processing, Vol. 28, pp. 199-219, 1984.

[13] M Unser,
"Sum and Difference Histograms for Texture Classification",
IEEE Trans. on Pattern Analysis and Machine Intell., Vol. PAMI-8, pp. 118-125, 1986.

[14] J.S. Weszka, C.R. Dyer, and A. Rosenfeld,
"A comparative Study of Texture Measures for Terrain Classification",
IEEE Trans. on Systems, Man and Cybernetics, Vol. SMC-6, pp. 269-285, 1976.

[15] JC-M. Wu, and Y-C. Chen,
"Statistical Feature Matrix for Texture Analysis",
Computer Vision, Graphics, and Image Processing; Graphical Models and Image Processing, Vol. 54, pp. 407-419, 1992.