ELSEVIER

Pattern Recognition Letters 20 (1999) 191–197

Pattern Recognition Letters

# Fast determination of textural periodicity using distance matching function

Gyuhwan Oh [a], Seungyong Lee [b], Sung Yong Shin [a,*]

[a] *Department of Computer Science, Korea Advanced Institute of Technology, Kusung-dong 373-1, Yusung-gu, Taejon 305-701, South Korea*

[b] *Department of Computer Science and Engineering, POSTECH, Pohang 790-784, South Korea*

Received 24 October 1997; received in revised form 6 November 1998

**Abstract**

The periodicity of a texture is one of its important visual characteristics. The inertias of co-occurrence matrices of the texture have been often used to detect the visual periodicity. However, it is time-consuming to explicitly construct these matrices. In this paper, we propose the distance matching function to avoid constructing the matrices due to our new interpretation of an inertia. For a texture of size $m \times n$, the inertias of all co-occurrence matrices can be obtained in $O(mn \log mn)$ time by simultaneously evaluating the function at all displacement vectors. This is a significant improvement over the previous method using the co-occurrence matrices, that requires $O(m^2n^2)$ time. 

*Keywords:* Texture; Patterns; Periodicity; Inertia; Co-occurrence matrix; Distance matching function

## 1. Introduction

The periodicity of a texture is one of its important visual characteristics. Indicating how textural patterns are repeated in the texture, it has been often used as a measure for texture discrimination at the structural level. A regular texture such as wallpaper can be generated by tiling its motif, that is, a minimal repeating region of the texture. Therefore, the motif gives a compact representation of the texture. Since the motif of a texture can be determined from its periodicity, knowledge of the periodicity is particularly useful for wallpaper design and compression (Stevens, 1991).

The conventional methods such as the Fourier transform and auto-correlation analysis may be employed to detect the periodicity of a texture of infinite size. Given such a texture, we can determine its periodicity from the distribution of impulses of its Fourier spectrum (Soliman and Srinath, 1990). For a periodic texture of finite size, a similar idea would be applied to detect its periodicity if the size were sufficiently large. In practice, periodic textures hardly meet this condition. Thus, the Fourier transform is

* Corresponding author. Tel.: 82 42 869 3528; fax: 82 42 869 3510; e-mail: syshin@jupiter.kaist.ac.kr

0167-8655/99/$ – see front matter 
PII: S 0 1 6 7 - 8 6 5 5 ( 9 8 ) 0 0 1 4 0 - 8

not suitable for detecting the textural periodicity. Similarly, auto-correlation analysis does not work, either, in such a case (Proakis and Manolakis, 1988).

It is well-known that second-order statistics play an important role in texture discrimination (Julesz, 1975). Each of these statistics represents a geometric relationship between every pair of points of a texture in a certain manner. A co-occurrence matrix has been commonly used to effectively obtain such statistics (Haralick, 1979; Tomita and Tsuji, 1990). There are three measures, based on co-occurrence matrices, such as $\chi^2$ statistics, $\kappa$ statistics and inertias, that have been used to identify the periodicity of a texture of finite size. The weakness of $\chi^2$ statistics has already been pointed out by Parkkinen et al. (1990): If there exists exactly one non-zero element on every row and column of a co-occurrence matrix, then its normalized $\chi^2$ value has the maximum value of unity. Thus, $\chi^2$ statistics fail to detect the textural periodicity in such a case. For $\kappa$ statistics and inertias, their ability to detect the textural periodicity has been shown by Parkkinen et al. (1990) and Conners and Harlow (1980), respectively. These methods first compute co-occurrence matrices for certain displacement vectors and then calculate their measures from these matrices. However, given a texture of size $m \times n$, it takes $O(m^2n^2)$ time to compute co-occurrence matrices for all displacement vectors. This seems unacceptable in practice if the texture size is large.

In this paper, we first propose a distance matching function that can replace the inertia of a co-occurrence matrix. Given a texture and a displacement vector, the function value of the texture with respect to the vector is equivalent to the inertia of its corresponding co-occurrence matrix. We then present an efficient method to evaluate the function. For a texture of size $m \times n$, all the function values can be computed in $O(mn \log mn)$ time. This is a significant improvement over $O(m^2n^2)$ time to evaluate the inertias of all co-occurrence matrices.

## 2. Distance matching function

A texture can be defined as a bivariate function $f$. The function value $f(x,y)$ represents the gray level at position $(x,y)$. Let $P_{\boldsymbol{v}}$ be the co-occurrence matrix with displacement vector $\boldsymbol{v}$. The element of $P_{\boldsymbol{v}}$ at $(s,t)$ is denoted by $P_{\boldsymbol{v}}[s,t]$. Given a displacement vector $\boldsymbol{v} = (v_x, v_y)$, $P_{\boldsymbol{v}}[s,t]$ gives the number of pixel pairs, $(u_x, u_y)$ and $(w_x, w_y)$, satisfying the following conditions: (i) $f(u_x, u_y) = s$ and $f(w_x, w_y) = t$, (ii) $u_x + v_x = w_x$ and $u_y + v_y = w_y$. The size of $P_{\boldsymbol{v}}$ is $Q \times Q$, where $Q$ is the number of gray levels in the texture. From the co-occurrence matrix, various kinds of texture features can be extracted to characterize the structure of the texture (Ballard and Christopher, 1982; Peckinpaugh, 1991; Tomita and Tsuji, 1990).

A texture $f$ is periodic with period $\boldsymbol{v} = (v_x, v_y)$ if and only if $f(x,y) = f(x+v_x, y+v_y)$ for all $x$ and $y$. Note that since $f$ has a finite size, the equality of $f(x,y) = f(x+v_x, y+v_y)$ is examined in the region where both $f(x,y)$ and $f(x+v_x, y+v_y)$ are properly defined. If $f$ is periodic with $\boldsymbol{v} = (v_x, v_y)$, then $P_{\boldsymbol{v}}$ is a diagonal matrix. Moreover, $P_{\boldsymbol{v}'}$ is also diagonal for any $\boldsymbol{v}' = (\alpha v_x, \beta v_y)$, where $\alpha$ and $\beta$ are integers (Conners and Harlow, 1980; Parkkinen et al., 1990; Zucker and Terzopoulos, 1980). In reality, textural patterns are hardly periodic because of their inherent noise. However, nonzero elements in $P_{\boldsymbol{v}}$ tend to be concentrated along the main diagonal for some $\boldsymbol{v}$ if the texture looks periodic. Therefore, co-occurrence matrices can also be used to detect the visual periodicity of noisy textures.

Let $P_{\boldsymbol{v}}$ be the co-occurrence matrix with displacement vector $\boldsymbol{v} = (v_x, v_y)$ over a texture $f$. Then, the inertia of $P_{\boldsymbol{v}}$ is defined as

$$I(P_{\boldsymbol{v}}) = \sum_{s=0}^{Q-1}\sum_{t=0}^{Q-1}(s-t)^2 P_{\boldsymbol{v}}[s,t], \tag{1}$$

where $Q$ is the number of gray levels in $f$ (Conners and Harlow, 1980; Peckinpaugh, 1991). The normalized inertia $I_{\mathrm{n}}(P_{\boldsymbol{v}})$ is obtained by dividing $I(P_{\boldsymbol{v}})$ with the sum of elements in $P_{\boldsymbol{v}}$. That is,

$$I_{\mathrm{n}}(P_{\boldsymbol{v}}) = \frac{1}{N}\sum_{s=0}^{Q-1}\sum_{t=0}^{Q-1}(s-t)^2 P_{\boldsymbol{v}}[s,t], \tag{2}$$

where $N = \sum_{s=0}^{Q-1}\sum_{t=0}^{Q-1} P_{\boldsymbol{v}}[s,t]$.

Notice that $P_{\boldsymbol{v}} = P^{\mathrm{T}}_{-\boldsymbol{v}}$ for all $\boldsymbol{v}$, where $P^{\mathrm{T}}_{-\boldsymbol{v}}$ is the transpose of $P_{-\boldsymbol{v}}$. Therefore, both $P_{\boldsymbol{v}}$ and $P_{-\boldsymbol{v}}$ have the same inertia value. This property makes it sufficient to consider only the case $\boldsymbol{v} = (v_x, v_y)$ for $v_y \geqslant 0$ in computing $I(P_{\boldsymbol{v}})$ and $I_{\mathrm{n}}(P_{\boldsymbol{v}})$.

In order to derive a new representation of Eq. (1), we first consider a one-dimensional texture $g$ of size $m$ and its co-occurrence matrix $P_v$. From Eq. (1), we have

$$I(P_v) = \sum_{s=0}^{Q-1}\sum_{t=0}^{Q-1}(s-t)^2 P_v[s,t]. \tag{3}$$

$P_v[s,t]$ can be rephrased by

$$P_v[s,t] = \sum_{i=0}^{m-v-1}\delta^v_{s,t}(i), \tag{4}$$

where

$$\delta^v_{s,t}(i) = \begin{cases} 1 & \text{if } g(i) = s \text{ and } g(v+i) = t, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

From Eqs. (3) and (4), we then obtain

$$I(P_v) = \sum_{s=0}^{Q-1}\sum_{t=0}^{Q-1}(s-t)^2\sum_{i=0}^{m-v-1}\delta^v_{s,t}(i) \tag{6}$$

$$= \sum_{s=0}^{Q-1}\sum_{t=0}^{Q-1}\sum_{i=0}^{m-v-1}|g(i) - g(v+i)|^2\delta^v_{s,t}(i) \tag{7}$$

$$= \sum_{i=0}^{m-v-1}|g(i) - g(v+i)|^2\sum_{s=0}^{Q-1}\sum_{t=0}^{Q-1}\delta^v_{s,t}(i). \tag{8}$$

Let a distance matching function $d$ for $g$ be defined as follows:

$$d(x) = \sum_{i=0}^{m-x-1}|g(i) - g(x+i)|^2. \tag{9}$$

Since $\sum_{s=0}^{Q-1}\sum_{t=0}^{Q-1}\delta^v_{s,t}(i) = 1$, we obtain

$$I(P_v) = d(v). \tag{10}$$

For example, consider $g = \{0,1,3,2,0,1,1,2,0,1,2,2\}$. Fig. 1(a) shows the contents of $P_1$ and the value of $I(P_1)$. For clarity, only nonzero elements are explicitly shown in $P_1$. In Fig. 1(b), $d(1)$ has the same value as $I(P_1)$ by computing the sum of squared differences between $g(i)$ and $g(1+i)$.

Since the sum of the elements in $P_v$ is $m-v$, the normalized inertia measure $I_{\mathrm{n}}(P_v)$ can be expressed by $(1/(m-v))d(v)$. That is, $I(P_v)$ and $I_{\mathrm{n}}(P_v)$ over a texture $g$ can be given by

$$I(P_v) = d(v) \quad \text{for } v = 0,\ldots,\left\lfloor\frac{m}{2}\right\rfloor, \tag{11}$$

$$I_{\mathrm{n}}(P_v) = \frac{1}{m-v}d(v) \quad \text{for } v = 0,\ldots,\left\lfloor\frac{m}{2}\right\rfloor, \tag{12}$$

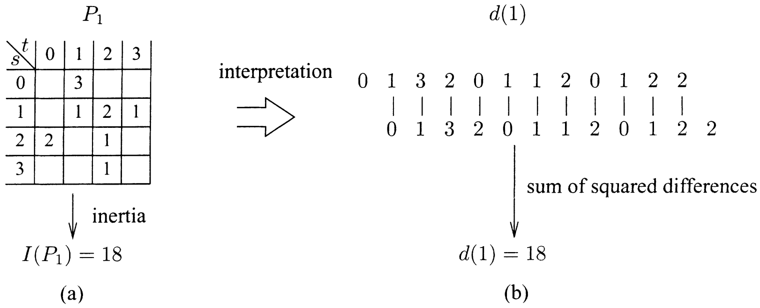


Fig. 1. Computation of $I(P_1)$ and $d(1)$ for $g = \{0, 1, 3, 2, 0, 1, 1, 2, 0, 1, 2, 2\}$.

respectively, where $m$ is the size of $g$. The upper bound of $v$ to compute $d(v)$ and $d_{\mathrm{n}}(v)$ is $\lfloor m/2 \rfloor$ because a period cannot exceed half of the size of a given one-dimensional texture.

Now, we extend this observation to the case of a two-dimensional texture $f$ of size $m \times n$. We define a two-dimensional *normalized distance matching function* $d_{\mathrm{n}}$ for $f$:

$$d_{\mathrm{n}}(x,y) = \begin{cases} d_{\mathrm{n}1}(x,y) & \text{for } x = 0,\ldots,\lfloor \frac{m}{2} \rfloor,\ y = 0,\ldots,\lfloor \frac{n}{2} \rfloor, \\ d_{\mathrm{n}2}(-x,y) & \text{for } x = 0,\ldots,-\lfloor \frac{m}{2} \rfloor,\ y = 0,\ldots,\lfloor \frac{n}{2} \rfloor, \end{cases} \tag{13}$$

where

$$d_{\mathrm{n}1}(x,y) = \frac{1}{(m-x)(n-y)} \sum_{i=0}^{m-x-1} \sum_{j=0}^{n-y-1} |f(i,j) - f(x+i, y+j)|^2, \tag{14}$$

$$d_{\mathrm{n}2}(x,y) = \frac{1}{(m-x)(n-y)} \sum_{i=0}^{m-x-1} \sum_{j=0}^{n-y-1} |f(m-1-i,j) - f(m-1-i-x, y+j)|^2. \tag{15}$$

As shown in Fig. 2, $d_{\mathrm{n}}$ computes the average of squared differences of pixel values over the region where the two copies of $f$ overlap.

$I_{\mathrm{n}}(P_{\boldsymbol{v}})$ with displacement vector $\boldsymbol{v} = (v_x, v_y)$ is $d_{\mathrm{n}1}(v_x, v_y)$ when $v_x \geqslant 0$ and $v_y \geqslant 0$. For $v_x \leqslant 0$ and $v_y \geqslant 0$, $I_{\mathrm{n}}(P_{\boldsymbol{v}})$ with $\boldsymbol{v} = (v_x, v_y)$ is $d_{\mathrm{n}2}(-v_x, v_y)$. That is,

$$I_{\mathrm{n}}(P_{\boldsymbol{v}}) = d_{\mathrm{n}}(v_x, v_y), \quad \text{for } v_x = -\lfloor m/2 \rfloor, \ldots, \lfloor m/2 \rfloor \text{ and } v_y = 0, \ldots, \lfloor n/2 \rfloor. \tag{16}$$

For a special case, if we detect the horizontal periodicity of $f$, then we have $\boldsymbol{v} = (v_x, 0)$. Accordingly, $d_{\mathrm{n}}$ can be simplified as

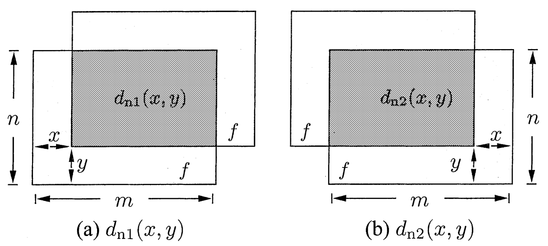


Fig. 2. Computation of $d_{\mathrm{n}}(x, y)$.

$$d_{\mathrm{n}}(v_x,0)=\frac{1}{(m-v_x)n}\sum_{i=0}^{m-v_x-1}\sum_{j=0}^{n-1}|f(i,j)-f(v_x+i,j)|^2,\ \ \text{for } v_x=0,\ldots,\left\lfloor\frac{m}{2}\right\rfloor. \tag{17}$$

Similarly, for the vertical periodicity, we have

$$d_{\mathrm{n}}(0,v_y)=\frac{1}{m(n-v_y)}\sum_{i=0}^{m-1}\sum_{j=0}^{n-v_y-1}|f(i,j)-f(i,v_y+j)|^2,\ \ \text{for } v_y=0,\ldots,\left\lfloor\frac{n}{2}\right\rfloor. \tag{18}$$

## 3. Efficient evaluation of all distance matching functions

We consider how to efficiently evaluate $d_{\mathrm{n}}(x,y)$ for all vectors $(x,y)$. $d_{\mathrm{n1}}(x,y)$ can be rearranged as

$$d_{\mathrm{n1}}(x,y)=\frac{1}{(m-x)(n-y)}(p(x,y)-2q(x,y)+r(x,y)),$$

where

$$p(x,y)=\sum_{i=0}^{m-x-1}\sum_{j=0}^{n-y-1}f^2(i,j), \tag{19}$$

$$q(x,y)=\sum_{i=0}^{m-x-1}\sum_{j=0}^{n-y-1}f(i,j)f(x+i,y+j), \tag{20}$$

$$r(x,y)=\sum_{i=0}^{m-x-1}\sum_{j=0}^{n-y-1}f^2(x+i,y+j). \tag{21}$$

Function $p$ can be expressed as

$$p(x,y)=p(x+1,y)+p(x,y+1)-p(x+1,y+1)+f^2(m-x-1,n-y-1), \tag{22}$$

for $x=0,\ldots,m-1$ and $y=0,\ldots,n-1$, where $p(m,\cdot)=0$ and $p(\cdot,n)=0$. Using Eq. (22), we can incrementally compute $p(x,y)$ for all $(x,y)$ starting from the top-right corner, $(m-1,n-1)$. In a similar way, all values of $r$ can be computed incrementally.

Function $q$ is the *autocorrelation* of $f$. From the *correlation theorem* (Gonzalez and Woods, 1992), it can be effectively computed in the frequency domain. That is,

$$q(x,y)=f(x,y)\circ f(x,y)\iff \mathscr{F}^*(u,v)\mathscr{F}(u,v), \tag{23}$$

where $\mathscr{F}$ is the Fourier transform of $f$ and $\mathscr{F}^*$ is the complex conjugate of $\mathscr{F}$. From Eq. (23), we can evaluate $q$ by computing the inverse Fourier transform of $\mathscr{F}^*\mathscr{F}$.

Now we consider how to compute $d_{\mathrm{n2}}(x,y)$. Let $f'(x,y)=f(m-x-1,y)$ (i.e., $f'$ is the vertical reflected texture of $f$). Therefore,

$$d_{\mathrm{n2}}(x,y)=\frac{1}{(m-x)(n-y)}\sum_{i=0}^{m-x-1}\sum_{j=0}^{n-y-1}|f'(i,j)-f'(x+i,y+j)|^2, \tag{24}$$

which has the same form as $d_{\mathrm{n1}}(x,y)$. Thus, we can compute $d_{\mathrm{n2}}(x,y)$ in the same way as that of $d_{\mathrm{n1}}(x,y)$. Eqs. (17) and (18) have simpler form than $d_{\mathrm{n1}}(x,y)$ and can be computed in a similar way.

We can incrementally compute $p(x,y)$ and $r(x,y)$ at all $(x,y)$ in $O(mn)$ time. $q(x,y)$ can be evaluated at all $(x,y)$ in $O(mn \log mn)$ time by using the fast Fourier transform (FFT). Therefore, it takes $O(mn \log mn)$ time to compute $d_{n1}(x,y)$ at all $(x,y)$. $d_{n2}(x,y)$ can also be evaluated within the same time bound. We can also compute Eqs. (17) and (18) in $O(mn \log m)$ and $O(mn \log n)$ time, respectively.

## 4. Experiments

We perform experiments to show how well our method works. We apply our method to the two textures, given in Fig. 3(a) and (c). These textures are scanned from the books by Grafton (1992) and Martin (1982). We compute inertias of co-occurrence matrices with displacement vectors $\boldsymbol{v} = (v_x, v_y)$, $v_x = -\lfloor m/2 \rfloor, \dots, \lfloor m/2 \rfloor$ and $v_y = -\lfloor n/2 \rfloor, \dots, \lfloor n/2 \rfloor$ for each texture, where $m \times n$ is the size of the texture. Fig. 3(b) and (d) gives plots of the inertias for the textures in Fig. 3(a) and (c), respectively. Each point in the former figures represents a displacement vector, and its intensity is set to be proportional to the inertia value for the vector. As observed in Fig. 3(b) and (d), the textures in Fig. 3(a) and (c) have orthogonal and non-orthogonal displacement vectors, respectively, that represent textural periods. Since the inertia values at those displacement vectors are locally minimal, we can extract such vectors by properly thresholding the inertia values. Our method can compute periods of textures in both orthogonal and non-orthogonal cases.

Fig. 4 shows plots of computation times for two measures, $\kappa$ statistics and inertias. The experiment is performed on a Silicon Graphics Indigo 2 workstation with an R10000 processor and 128 Mbytes RAM.

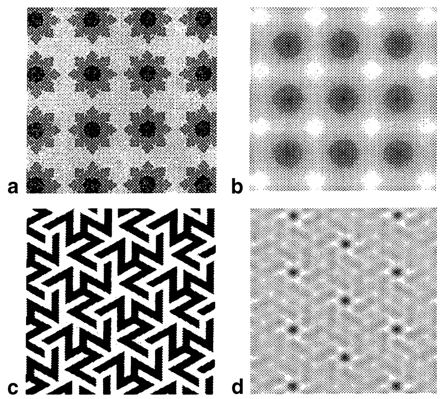


Fig. 3. Two textures and plots of each of the inertias. (a) A texture. (b) Plot of inertias. (c) A texture. (d) Plot of inertias.

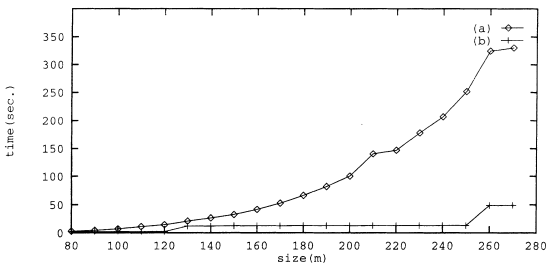


Fig. 4. Time comparison: $\kappa$ statistics (a) computed from co-occurrence matrices and inertias; (b) computed using the distance matching function.

For a texture of size $m \times m$, we compute $\kappa$ statistics of co-occurrence matrices for all displacement vectors after quantizing the texture to have four gray levels. On the other hand, we compute inertias of the same texture using the distance matching function. We measure each of their computation times in seconds. As $m$ increases, the computation time of $\kappa$ statistics grows rapidly, while that of inertias grows very slowly. The time for computing $\chi^2$ statistics shows the same behavior as that for computing $\kappa$ statistics because the time for constructing co-occurrence matrices dominates the rest of computation time. The experimental results support that our proposed method outperforms co-occurrence matrix-based methods.

## 5. Conclusion

It is time-consuming to determine the textural periodicity using co-occurrence matrices. In this paper, we first proposed a distance matching function that can be driven directly from a texture. Given a displacement vector, the function value with respect to the vector is equivalent to the inertia of the corresponding co-occurrence matrix. Employing the correlation theorem (Gonzalez and Woods, 1992), we then showed that this function can be evaluated at all positions over a texture in $O(mn \log mn)$ time. This is a significant improvement over computing inertias of all co-occurrence matrices, which requires $O(m^2 n^2)$ time.

## References

Ballard, D.H., Christopher, M.B., 1982. Computer Vision. Prentice-Hall, Englewood Cliffs, NJ, pp. 186–188.

Conners, R.W., Harlow, C.A., 1980. Toward a structural analyzer based on statistical methods. Computer Graphics and Image Processing 12, 224–256.

Gonzalez, R.C., Woods, R.E., 1992. Digital Image Processing. Addison-Wesley, Reading, MA, pp. 109–111.

Grafton, C.B., 1992. Decorative Tile Designs in Full Color. Dover, New York, p. 54.

Haralick, R.M., 1979. Statistical and structural approaches to texture. Proc. IEEE 67, 786–804.

Julesz, B., 1975. Experiments in the visual perception of texture. Science American 232, 34–43.

Martin, G.E., 1982. Transformation Geometry. Springer, Berlin, p. 42.

Parkkinen, J., Selkäinaho, K., Oja, E., 1990. Detecting texture periodicity from the cooccurrence matrix. Pattern Recognition Letters 11, 43–50.

Peckinpaugh, S.H., 1991. Note: An improved method for computing gray-level co-occurrence matrix based texture measures. Graphical Models and Image Processing 53 (6), 574–580.

Proakis, J.G., Manolakis, D.G., 1988. Introduction to Digital Signal Processing. Macmillan, New York, pp. 246–276.

Soliman, S.S., Srinath, M.S., 1990. Continuous and Discrete Signals and Systems. Prentice-Hall, Englewood Cliffs, NJ, pp. 352–354.

Stevens, P.S., 1991. Handbook of Regular Patterns – An Introduction to Symmetry in Two Dimensions. MIT Press, Cambridge, MA, pp. 168–331.

Tomita, F., Tsuji, S., 1990. Computer Analysis of Visual Textures. Kluwer Academic Publishers, Dordrecht, pp. 13–36.

Zucker, S.W., Terzopoulos, D., 1980. Finding structure in co-occurrence matrices for texture analysis. Computer Graphics and Image Processing 12, 286–308.