COMPARISON OF SOME TEXTURE CLASSIFIERS

Einari Kilpelä and Jan Heikkilä
Helsinki University of Technology
Institute of Photogrammetry and Remote Sensing
02150 Espoo 15, Finland

ABSTRACT

A performance analysis between different textural feature descriptors in land-use classification is presented. Both satellite and aerial images are used.

The texture descriptors used are first order statistics, second order (cooccurrence) statistics, Fourier spectrum, amplitude varying rate statistics and fractal descriptors. The technical implementation of each of these descriptors in the context of classification is also addressed. Each data set is classified using the spectral features, each of the texture descriptors and some variations of them, and using a combination of spectral and textural features. The classifiers used are the maximum likelihood classifier assuming multinormal density functions, the k-NN classifier and the average learning subspace (ALSM) classifier.

The performance analysis, which is based on independent test sites, shows that the ALSM-classifier and the k-NN classifier work equally well, but the crude assumption of normal densities in the context of maximum likelihood classifier produces biased results. No clear distinction between the behavior of the different texture descriptors was found. The full usage of the cooccurrence statistics works well. However, its computational load is quite heavy. The more simple texture descriptors, like the simple fractal dimension in combination with spectral features, works often equally well in the context of satellite images. In case of larger scale images, the more complex texture descriptors are called for.

1. INTRODUCTION

Texture is an important cue for understanding and discriminating in natural images. However, it is surprisingly seldom utilized in the context of terrain classification. In many applications (e.g. land-use classification) texture features could bring more discriminatory information. This is especially true when larger scale imagery is used. When computing textural feature vectors for each pixel according to some local neighborhood, a little bit of the rude assumption of spatial independence can be broken down. This does not mean that one should give up from the attempts to more properly model the sampling process, e.g. with the Markov random field models (see /GemGem84/) and further develop their computational characteristics especially for multi-dimensional spaces. This should be the final goal. In this paper we are anyhow concerned with more conservative and practical approaches.

The problem of texture analysis and modelling is a widely discussed problem in the areas of Pattern Recognition, Image Analysis, Computer Vision and even in Computer Graphics. Texture is a commonly used criteria in the early processing of visual information. Paradoxically however, because of its loose definition, a huge amount of methods, both ad hoc and formal, have been developed (for surveys see /Harali79/, /GoDeOo85/ and /Harali86/). The methods fall into two main categories, namely statistical and structural. The naming convention is slightly misleading, because usually quite a lot of statistics is involved in the structural approaches, too. Images taken over natural terrain contain both spatially and spectrally quite irregularly distributed, usually microscopic, texture elements. The smaller the imaging scale, the less structure it has. In many circumstances, just a simple measure of the roughness of the texture can bring enough discriminatory power to the feature space. Anyhow, the larger the scale, the more structure is visible in the texture. Excluding manmade objects, the spatial distribution of the (maybe invisible) textural structure elements is usually quite irregular also in large scale (aerial) images. Due to these facts, statistical methods are preferred when analyzing textures in natural images. So is the case also in the underlying project.

Because of the variability of the texture measures, a practitioner faces the problem of choosing the most suitable descriptor for his application. Reviewing the literature does not help much, because no thorough comparison exists. There are so many factors which influence the performance of a texture classifier (the data, the texture descriptor, the number of features, the type of classifier, the number of training samples, resolution level, preprocessing steps etc.) that a complete comparison would be a huge task. There are some texture measures, which have been quite successful in single comparative studies and which have become quite popular. One of the most popular texture descriptors is the second order statistics (Cooccurrence Statistics), originally suggested by Haralick, Shanmugam and Dinstein in 1973 (/HaShDi73/). Another, widely used descriptor is the Fourier power spectrum. These two methods are compared in many

papers to each other, but contradictory results have been achieved. This certainly comes from the difficulty involved in their usage and parameter tuning. Both methods, as such, produce high dimensional feature vectors and the usual approach is to compute some ad hoc features from the original descriptors. This reduces the original information and makes the comparison difficult. We have tried to avoid this problem by careful parameter tuning and by utilizing standard feature extraction methods (/DevKit82/) which do not reduce drastically the amount of information but only the dimensionality of the feature vectors. In addition to these two popular texture descriptors, we have included in the comparison three other texture measures. Firstly, a simple first order statistic in the form of local variance, serves as a kind of reference. Secondly, the appealing fractal based descriptors, the fractal dimension and the fractal signature, are included. Thirdly, a new method, called the amplitude varying rate statistical approach after Zhuang and Dunn /ZhuDun90/, is included in the comparison, because of the most promising results achieved in /ZhuDun90/.

When comparing the performance of texture descriptors in the context of classification, attention has to be paid, not only to the descriptors, but also to the classifier itself. It has to be chosen to properly work with the features chosen. The usual brute-force application of an "optimal" maximum likelihood classifier assuming multi-normal probability densities, has been, for the writers' opinion, distorting many comparative studies. Especially, when using textural descriptors, the decision boundaries can be highly non-linear. In these instances, a non-parametric classifiers would be the only reasonable choice. A simple, but computationally heavy, k-NN classifier, has been proven to have a large sample size error rate that decreases monotonically to the optimal Bayesian error rate /CovHar67/. Its computational complexity can also be thoroughly improved by the so called editing and condensing techniques (see Chapter 3). Because a k-NN classifier can produce highly non-linear decision boundaries, it is extensively compared with the ML-classifier in the present paper. The non-linearity problem is widely addressed in Artificial neural network classifiers. Such an adaptive classifier is the Average Learning Subspace Method (ALSM) developed by Oja in /Oja83/. Because of its reported suitability to texture classification, especially in the context of cooccurrence statistics and power spectral methods, it is the third classifier adopted in this context.

In Chapter 2 we will review the texture descriptors, their technical implementation, and the feature extraction methods utilized in this project. Chapter 3 is concentrated on the description of the classifiers used, and Chapter 4 gives a summary of the results. Finally Chapter 5 draws some conclusions.

2. TEXTURE DESCRIPTORS

Haralick defines texture as consisting of two basic dimensions /Harali79/. The first one consists of the image texture elements itself, and the second one of the spatial dependencies between these elements. This spatial organization may be random or may have dependencies between its primitives. This dependence may be structural, probabilistic, or functional. Texture can be described with such words as fine, coarse, smooth, granular, regular, irregular, random, or structural.

Even today there is no exact mathematical definition of texture and we still rely on those loose descriptions. A large part of the texture analysis techniques are in fact ad hoc and many statistical approaches to the measurement and characterization of image texture exist. In statistical methods, the pixels are supposed to have spatial distribution having some statistical characteristics and the analysis techniques try to determine corresponding parameters. The statistical characteristics, which one measures, make the difference between the methods. For a good survey see e.g. /Harali79/, /Harali86/, or /GoDeOo85/.

In the underlying comparison we have chosen four texture descriptors which are reported to own good discriminative characteristics, namely the second order cooccurrence statistics /HaShDi73/, the 2D power spectrum /Bajcsy73/, the fractal descriptors /Pentla83/, and the amplitude varying rate approach /ZhuDun90/. These methods, the problems involved and their technical implementation will be addressed in Chapters 2.1-2.4.

2.1 Second order (cooccurrence) statistics

There has been psychovisual evidence that two textures with identical second order statistics are not separable from each other /Julesz62/. Later it was pointed out by Gagalowicz and Tournier-Lasserve, that for non-homogeneous textures this does not hold /GagTou86/. Gagalowicz and Tournier-Lasserve also claim that natural textures are usually inhomogeneous. However, in practice it seems to be a good approximation for texture distinguishability.

The cooccurrence matrix (often referred as the Gray Tone Spatial Dependence Matrix) is an estimate of the second order joint conditional probability density function, and is defined by /DyHoRo80/ as follows:

*Cooccurrence matrix is a G*G matrix, where each entry (i,j) is the number of times gray levels i and j occur at separation d in the picture, which has been quantized to G levels.*

The high dimensionality of the cooccurrence matrix produces the first problem. In 8 bit images, a straightforward application

would produce 256*256 dimensional feature vectors. However, the estimates would be highly unreliable. The normal procedure to compress the number of gray levels, leads to 8*8 or 16*16 dimensional vectors, which are still quite high. In the classical paper of /HaShDi73/ a set of fourteen features are derived from the cooccurrence matrix. Usually only a few of them have been used, the famous five being energy, entropy, contrast, correlation, and homogenity. However, the features that Haralick et al. extract from the cooccurrence matrix, reduce the amount of information. E.g., Conners and Harlow /ConHar80/ demonstrated by two different textures, having different cooccurrence matrices, that the five features were the same, even when using different separation parameters d.

Often, the use of just one separation parameter is not sufficient. For each separation parameter, its own matrix can be computed and the features extracted can be averaged or concatenated. We have chosen to use the cooccurrence matrix as a feature vector, without computing any ad hoc features out of it. To reduce the dimensionality, we have used normal feature extraction methods (see e.g. /DevKit82/), and transformed the autocorrelation matrix to an orthogonal subspace. This strategy was originally applied in /OjaPar87/.

The second problem comes from the parameter tuning; which are the separation parameters to be chosen for our purpose? Zucker and Terzopoulos considered the cooccurrence matrix as a contingency table, and used χ-statistics to analyze periodicities in the texture for finding the right separation parameter /ZucTer80/. However, this statistics have been criticized by Selkäinaho, Parkkinen and Oja (/SePaOj87/ and /SePaOj88/). They demonstrated that this statistics does not properly discriminate among the types of dependencies indicated by the cooccurrence matrix. They also suggested a new statistic, the κ-statistics, which they demonstrated to work better than the χ-statistics. Also the computational complexity is much lower. We have applied their methodology in tuning the parameters of our texture descriptors.

The following algorithm summarizes our textural feature extractor in the case of cooccurrence statistics:

(1) For each class, compute all possible (using all possible separation parameters suitable for the window size) cooccurrence matrices, form their κ-statistics, and take the one with highest value to present the cooccurrence statistics of that class. Compress the number of gray levels to 8 (dimension of the corresponding feature vector is 64).

(2) Combine the resulting matrices to a feature vector, compute the autocorrelation matrix, perform a KL-transformation (See /DevKit82/) to the resulting matrix, and take the part which statistically describes 99% of the information to represent the cooccurrence features. For the ALSM-classifier this is done separately for each class.

2.2 The power spectral method

The power spectrum of the 2D Fourier transformation is another widely used textural descriptor. Since specific components in the frequency domain representation contain explicit information about the spatial distribution, useful features are obtained. It was first applied by Bajcsy /Bajcsy73/, who derived several features from the spectrum and showed its power in the problem of texture analysis. The traditional texture features, after Weszcka et al. /WeDyRo76/, extracted from the spatial frequency domain are usually limited to an array of summed spectral energies within ring and wedge shaped regions. This results in a good texture discrimination only, if the chosen ring or wedge energies happen to be measured from correct locations of the power spectrum. D'Astous and Jernigan /DasJer84/ used a more intelligent methodology by measuring the distributions of the frequency components. They concluded to use five descriptors for each peak in the power spectrum and three global measures for the whole spectrum. Later Liu and Jernigan tried to find "still better texture measures" from the Fourier domain /LiuJer90/. They extracted a total of 28 features from the power spectrum and from the phase spectrum.

We have used here the same strategy as we used in the context of the second order statistics. The whole power spectrum is used as such, without any special feature extraction. For small windows the dimensionality of the spectrum is quite low, and the final reduction is done with the help of the orthogonal transformation. For comparison we have included the method described in /DasJer84/.

2.3 The fractal descriptors

The appealing concept of fractals by Mandelbrot /Mandel77/ has also been applied to the problem of texture analysis. A theoretical fractal object is self-similar to all magnifications, meaning that each segment of the object is statistically similar and invariant over scale transformations. The only description of the metric properties of an ideal fractal comes from the fractal dimension, which is usually higher than the topological dimension. The applicability in texture analysis is due to the fact, that the fractal dimension of a surface corresponds quite closely to our intuitive notion of roughness. The more wiggling is the object, the higher fractal dimension it has.

Most real world objects are not ideal fractals (e.g. /Goodch80/). Instead, the fractal dimension varies along scale. Rather than using the fractal dimension in the strict sense, the changes of the fractal dimension can be registered along the scale. This should give more power to the

descriptor and was utilized by Peleg et al. /PeNaHa84/ in the context of texture classification. Peleg et al. called this descriptor the factal signature. They demonstrated the power of the method in the context of some Brodatz textures.

The technical problem of measuring the fractal dimension of surfaces is quite difficult. Recently, Roy et al. showed empirically that dramatically different dimensions (2.01-2.33) can be achieved by applying different algorithms to the same data /RoGrGa87/. Most of the methods utilized are based on the so-called variogram approach, which can be performed both in the spatial and frequence domains (see e.g. /Pentla83/). This shows that the fractal descriptor indeed has some similarities with the other two methods presented above (using Fourier spectra or second order statistics). Other methods utilize e.g. the number of cubes that are necessary to estimate the volume /NaSoTa87/, the surface area covered by a blanket /PeNaHa84/ etc.

The approach we have used originates from the classical problem of measuring the length of a coastline. In a texture window, the length of specific profiles are computed at each scale. According to the self-similar properties of fractals, $[N(e)*e^{D} = C]$ should hold at each scale. Here e is the scale, N(e) the number of units needed and C is a constant. By taking the logarithm over this equation, parameters C and D can be estimated with a least-squares procedure. The estimation is done separately in each of the four main directions at 10 successive scales. This produces either 4- (fractal dimension) or 32 descriptors (in case of fractal signatures). In each direction, three profiles are used and the final value is an average of these three profiles. The fractal signatures are always computed with the help of three successive scales, similarly to the works of /PeNaHa84/ and /NaSoTa87/.

2.4 The amplitude varying rate approach

Very recently, Zhuang and Dunn reported for a new texture measure, which they called the amplitude varying rate approach /ZhuDun90/. In their method the Amplitude Varying Rate Matrix (AVRM) is computed through examining the profile of each scan line in a fixed direction and recording frequencies of distances between pixels with the same gray level. From this matrix they are able to estimate the sizes of the primitives and the periodicity and contrast of textures. Zhuang and Dunn strongly argued that their method is better than the cooccurrence matrix method, because it can describe some physical interpretations. They also showed empirically that their algorithm works better than the second order statistics. The result can be made questionable, because only the Haralick's five most popular features where used.

Because of these promising results we wanted to include this descriptor to the test. Again the AVRM-method was utilized in the same framework as the first two methods reported.

3. CLASSIFIERS

Because the usual assumption of multi-normal probability density functions in the context of parametric classifiers does not hold in texture classification, the Bayesian optimality of such a classification system is brutally violated. That is why, the maximum likelihood (ML) classifier serves here just as a reference. The other two classifiers (k-NN and ALSM) applied are both non-parametric and can better adopt themselves to the non-linear decision boundaries.

3.1 The k-NN classifier

The k-NN classifier (see e.g. /DevKit82/) can be regarded as the most important classifier with respect to practical applications. It has been proven in /CovHar67/ that the (large sample) error rate of the k-NN classifier monotonically decreases towards the optimal error bound of a Bayesian classifier as k goes towards infinity. When sample size is finite, this is not anymore valid /Devivj80/.

The proper choice of k is of course a difficult problem. In principle, one should choose k as big as possible, but practical problems will occur, because of the finite sample sizes (k does not monotonically decrease the classification accuracy). A useful guideline given in literature suggests to select k proportional to the square root of the sample size.

The k-NN classifier has not been too widely used in practical applications, because of the storage and computational complexity it imposes. However, techniques have been presented for competing with traditional techniques in this respect. The solution is to use two preprocessing techniques, namely, Editing and Condensing (see /DevKit80/). The idea is to select a small subset from the training set such that the 1-NN classification with the reduced dataset achieves a performance, which is close to or better than the performance of 1-NN classification with the complete set. The editing procedure is based on the holdout technique and can be summarized as follows /DevKit80/:

(1) Make a random partition of the available training data into N subsets (*diffusion*).

(2) Classify the samples in subset i using the k-NN of subset MOD((i+1),N) (*classification*).

(3) Discard all the samples that were misclassified at step 2 (*editing*).

(4) Pool the remaining data to constitute a new data set (*confusion*).

After this procedure has finished the final classification happens using the resulted dataset and the 1-NN classifier. By the so called Multiediting approach, the number of samples can be further reduced by repeating the above editing procedure until no editing occurs. In /DevKit80/ both theoretical and experimental evidence is given for proving the most favorable result, that the above procedure converges asymptotically (in the number of iterations) to the Bayes decision rule.

After the multiediting algorithm, the data is nicely clustered (because all the misclassified samples are rejected). The 1-NN classifier creates piecewise linear decision boundary (being approximation of the Bayes boundary). This boundary is actually defined by a small subset of samples belonging to the outer boarders of the clusters. It is clear that only these sample points are needed for the 1-NN classifier. The aim of *condensing* is to find such representative points.

So after multiediting and condensing, a mostly powerful 1-NN classifier can be used. It has the property of approximating the Bayes decision boundary and being very fast to compute. Our k-NN classifier has been implemented by using this technique.

3.2 The Average Learning Subspace Method

The subspace methods are reported in /Oja83/. A subspace in the n-dimensional pattern space is spanned by p linearly independent, orthonormal basis vectors a_i. The dimension of the subspace is then p. From the point of view of classification, a subspace restricts the possible directions in it, while the lengths remain undetermined. Suppose we have M classes in the classification problem, each being represented by a subspace, with dimensionalities p_i. Usually p_i is much lower than the original dimension. The simple classification rule then states that: if the distance between the pattern vector x and subspace i is smaller than between the pattern vector and subspace j, then classify x in class i. The distance from the subspace can be computed as follows:

$$d(x,L) = |x|^2 - \Sigma_p (x^T u_i)^2,$$

where u_i are the orthonormal basis vectors.

Since $|x|^2$ is the same for each class, it can be dropped and the classification rule consists only of inner products. So, if the subspaces dimensions are small, the classifier is very fast.
The essential question in the subspace method is how to actually construct the class subspaces to obtain optimal performance. The CLAFIC method /Watana69/ forms each subspace by minimizing the mean square error of the distances of a training set. This can be shown to be equivalent of maximizing

$$\Sigma_p \ (u_j{}^T C u_j),$$

where C is the autocorrelation matrix. The eigenvalue decomposition solves this problem and usually the first few eigenvectors span the subspace.

A serious drawback of the CLAFIC method is that each class subspace, although depending on the statistics of the class, is formed independently from the other classes. So, two classes overlapping each other may be very hard to discriminate. This leads to the ALSM-method which adapts itself better to this situation by learning. In the ALSM-method, the autocorrelation matrices are updated according to misclassifications. This means that, if either a sample vector of class i is misclassified to another class (j), or a sample vector of another class (k) is misclassified to i, the learning phase of the classifier updates the autocorrelation matrices of each class by:

$$R_m = R_{m-1} + \Sigma\ \alpha * S_m^{(i,j)} - \Sigma\ \beta * S_m^{(k,i)}.$$

Here α and β are small constants, and should be small enough to avoid overshooting. In this learning phase, the subspaces will be iterated sufficiently long so, that the classification of the training set becomes stable. The algorithm can be proven to converge /Oja83/. The constants α and β were both set to values 0.005 in this project.

4. SUMMARY OF THE RESULTS

The test material consists of two 1:15 000 aerial images both digitized in 4500*4500 format, one SPOT scene from rural and one SPOT scene from urban area. Two independent test sites were created. The other was used as a training sample and consisted of approximately 1300 samples for each class. The other was used as an independent test set and consisted of approximately 5000 samples for each class. The classification tried to separate five classes, urban or residential areas, two types of forest areas, pasture land or parks, and fields. In case of multichannel imagery, the textural descriptors were computed from channel 3 (infrared).

The practical classification was performed using a window of 31*31 pixels in the aerial images and using a window of 13*13 in the SPOT images.

Subspaces vs. extracted ad hoc features

Some preliminary runs were made to have an idea, if there is some difference in using the traditional feature descriptors of the cooccurrence statistics and the Fourier power spectrum, compared to the strategy of using these descriptors as such. These tests were performed on the basis of the aerial images only.

The dimensional reduction achieved by an orthogonal transformation was surprisingly high. The dimension of the final subspace was usually 5-10, although the original

dimensionality was at least 256 (16*16). Still the discriminatory power of the transformed space was better or the same than by the five features of Haralick. The same applies for the power spectrum method compared to the descriptors of /Das-Jer84/. This is only true for the k-NN and ALSM classifiers, but not for the ML-classifier, showing again the problems involved in the ML-classification. In that case the performance of the classifier distorted the results so, that correct conclusion could not be made.

Descriptor	Classifier		
	ALSM	k-NN	ML
Variance	73/78	77/78	65/68
Cooccurence	97/98	98/98	76/75
Fourier Spectrum	93/98	94/96	75/77
Fractal-dimension	76/89	75/90	69/80
Fractal-signature	76/88	76/90	69/74
AVHR	94/96	91/95	77/78

Table 1. Summary of the results in case of 1:15000 aerial imagery. Precentage of correct classifications, without/with spectral features.

Descriptor	Classifier		
	ALSM	k-NN	ML
Variance	59/82	59/82	55/61
Cooccurence	86/97	87/98	69/82
Fourier Spectrum	74/94	74/93	65/77
Fractal-dimension	71/94	71/95	69/81
Fractal-signature	70/95	71/94	63/74
AVHR	80/93	81/93	70/74

Table 2. Summary of the results in case of the SPOT image 1 (rural). Precentage of correct classifications, without/with spectral features.

Descriptor	Classifier		
	ALSM	k-NN	ML
Variance	62/81	62/82	59/69
Cooccurence	87/96	87/97	70/86
Fourier Spectrum	71/92	73/92	65/82
Fractal-dimension	71/94	71/95	69/81
Fractal-signature	70/95	70/94	63/80
AVHR	80/93	81/93	70/74

Table 3. Summary of the results in case of the SPOT image 2 (urban). Precentage of correct classifications, without/with spectral features.

Comparison of the descriptors

Tables 1-3 summarize all the results. As can be seen no clear distinction can be made, but usually the cooccurrence statistics yield the best results, achieving a very low error rate of 2-4%. As could be predicted, the larger scale imagery favors the more complex descriptors. At smaller scale images, also the very simple fractal descriptor produces good results (error rate of 7%) and it clearly competes the other simple descriptor, namely the variance.

Against the expectations, the fractal signature does not bring more information as compared to the fractal dimension. This might be caused by the relatively small window size, together with the method of estimating the fractal signature.

Contradictionary to the conclusion drawn in /ZhuDun90/, the AVHR seems to own a little bit higher error rate than the cooccurrence statistics.

In the SPOT images, it seems clear, that the textural descriptors alone cannot bring satisfactory results.

Comparison of the classifiers

As could be expected the ML-classifier behaves worst. No separation between the performance of the k-NN and ALSM-classifiers can be seen.

5. CONCLUDING REMARKS

The results indicated very clearly that the choice of a classifier is utmost important, when texture classification is performed. Both non-parametric classifiers used (k-NN and ALSM) can be highly recommended in this context. The usage of a ML-classifiers should be avoided.

For larger scale imagery some more complex measures are asked for, but in case of smaller case images, the simple descriptors based on computed fractal dimension in four main directions of a local window seem to work nicely and are computationally light. In the case of satellite images, the spectral channels should be combined to the texture descriptors before reasonable results can be expected.

The reason for these relatively optimistic results (error rates in the order of 5%) comes partly from the test data. Only ideal windows were used. In practical applications, the boarder areas of the texture areas cause some troubles and this test should be carried out also by using such indistinct areas.

Although the result show promising out, one should not forget, that the methods applied, are all rather heuristic in nature. The best way for texture analysis should be to model the whole sampling process, e.g. with the help of stochastic 2D processes. We hope that in the future the algorithms and hardware implementations are powerful enough to utilize these more complete and more formal models.

6. REFERENCES

/Bajcsy73/ Bajcsy, R.: Computer Description of Textured Surfaces. 3rd International Joint Conference on Artificial Intelligence, 1973, Stanford, pp. 572-579.

/ConHar80/ Conners, R.W., Harlow, C.A.: A Theoretical Comparision of Texture Algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 2, No 3, 1980, pp. 204-222.

/CovHar67/ Cover, T.M., Hart, P.E.: Nearest Neighbour Pattern Classification. IEEE Transactions of In-

formation Theory, Vol 13, January 1967, pp. 21-27.

/DasJer84/ D'Astous, F., Jernigan, M.E.: Texture Discrimination Based on Detailed Measures of the Power Spectrum. Proceedings of the seventh International Conference on Pattern Recognition, Montreal 1984, pp. 83-86.

/Devivj80/ Devivjer, P.A.: An Overview of Asymptotic Properties of Nearest Neighbour Rules. In Pattern Recognition in Practice, (Gelsema, E.S., Kanal, L.N. eds.), North-Holland, 1980, pp. 343-350.

/DevKit80/ Devivjer, P.A., Kittler, J.: On the Edited Nearest Neighbour Rule. Proceedings of the fifth international Conference of Pattern Recognition, Miami Beach 1980. pp. 72-80.

/DevKit82/ Devivjer, P., Kittler, J.: Pattern Recognition: A Statistical Approach. Prentice-Hall 1982.

/DyHoRo80/ Dyer, C.R., Hong, T.H., Rosenfeld, A.: Texture Classification Using Gray Level Co-occurrence Based on Edge Maxima. IEEE Transactions on Systems, Man, and Cybernetics, Vol 10, No 3, 1980, pp. 158-163.

/GagTou86/ Gagalowicz, A., Tournier-Lasserve, C.: Third-Order Model for Non-homogeneous Natural Textures. Proceedings of the eight International Conference on Pattern Recognition, Paris 1986, pp. 409-414.

/GemGem84/ Geman, S., Geman, D.: Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 6, 1984, pp. 721-741.

/GoDeOo85/ van Gool, L., Dewaele, P., Oosterlinck, A.: Texture Analysis Anno 1983. Computer Vision, Graphics and Image Processing, Vol 29, 1985, pp. 336-357.

/Goodch80/ Goodchild, M.F.: Fractals and the Accuracy of Geographical Measures. Mathematical Geology, Vol 12, pp. 85-98.

/Harali79/ Haralick, R.: Statistical and Structural Approaches to Texture. IEEE Proceedings, Vol 67, No 5, 1979, pp. 304-322.

/Harali86/ Haralick, R.: Statistical Image Texture Analysis. In Handbook of Pattern Recognition (Young, T.Y., Fu, K-S. Eds.), Academic Press 1986, pp. 247-280.

/HaShDi73/ Haralick, R., Shanmugam, K., Dinstein, I.: Textural Features for Image Classification. IEEE Transactions on Systems, Man, and Cybernetics, Vol 3, No 6, 1973, pp. 610-621.

/Julesz62/ Julesz, B.: Visual Pattern Discrimination. IRE Transactions on Information Theory, Vol 8, 1962, pp.84-92.

/LiuJer90/ Liu, S-S, Jernigan, M.E.: Texture Analysis and Discrimination in Additive Noise., Computer Vision, Graphics and Image Processing, Vol 49, 1990, pp. 52-67.

/Mandel82/ Mandelbrot, B.B.: The Fractal Geometry of Nature. Freeman 1982.

/NaSoTa87/ Meteorological Satellite Image Analysis Using Fractal Dimension and Lower-order Statistics. Proceedings of the fifth Scandinavian Conference on Image Analysis, Stockholm 1987, pp. 261-267.

/Oja83/ Oja, E.: Subspace Methods of Pattern Recognition. Research Studies Press, Letchworth and Wiley, 1983.

/OjaPar85/ Oja, E., Parkkinen, J.: Texture Classification with the Subspace Method. Proceedings of the fourth Scandinavian Conference on Image Analysis, Trondheim 1985, pp. 429-436.

/OjaPar87/ Oja, E., Parkkinen, J.: Texture Subspaces. In Pattern Recognition Theory and Applications (Devivjer, P., Kittler, J. eds.), Springler 1987.

/PeNaHa84/ Peleg, S., Naor, J., Hartley, R., Avnir, D.: Multiple Resolution Texture Analysis and Classification. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 6, No 4, 1984, pp. 518-523.

/Pentla83/ Fractal Based Description of Natural Scenes. Proceedings of IEEE Computer Vision and Pattern Recognition meeting, Arlington 1983, pp. 201-209.

/RoGrGa87/ Roy, A.G., Gravel G., Gauthier, C.; Measuring the Dimension of Surfaces: A Review and Appraisal of Different Methods. Proceedings of the eighth International Symposium on Computer Assisted Cartography, Baltimore, 1987, pp. 68-77.

/SePaOj87/ Selkäinaho, K., Parkkinen, J., Oja, E.: Structure in Cooccurrence Matrices: Association vs. Agreement. Proceedings of the fifth Scandinavian Conference on Image Analysis, Stockholm 1987, pp. 411-417.

/SePaOj88/ Selkäinaho, K., Parkkinen, J., Oja, E.: Comparision of χ^2 and κ Statistics in Finding Signal and Picture Periodicity. Proceedings of the ninth Conference of Pattern Recognition, Rome 1988, pp. 1221-1224.

/Watana69/ Watanabe, S.: Knowing and Guessing - a Quantitative Study of Inference and Information. Wiley 1969.

/WeDyRo76/ Weszka, J.S., Dyer, C.R., Rosenfeld. A.: A Comparative Study of Texture Measures for Terrain Classification. IEEE Transactions on Systems, Man and Cybernetics, Vol 6, No 4, 1975, pp. 269-285.

/ZhuDun90/ Zhuang, C., Dunn, S.: The Amplitude Varying Rate Statistical Approach for Texture Classification. Pattern Recognition Letters, Vol 11, February 1990, pp. 143-149.

/ZucTer80/ Zucker, S.W., Terzopoulos, D.: Finding Structure in Cooccurrence Matrices for Texture Analysis. Computer Graphics and Image Processing, Vol 12, 1980, pp. 286-308.