



ANALYSIS AND SYNTHESIS OF TEXTURES: A CO-OCCURRENCE-BASED APPROACH

GABRIELE LOHMANN

Technische Universität München, Institut für Informatik, Orleansstr. 34, 81667 München, Germany,
e-mail: lohmann@informatik.tu-muenchen.de

Abstract—The selection of suitable features is the most critical part of any classification process. In the area of textural classification, one way to determine if the features are suitable is to synthesize an image which has the given textural features. An algorithm for synthesizing textures that have a set of given co-occurrence matrices is presented, and it is shown that these synthetic images do indeed match their real counterparts very closely.

The successful synthesis of textures motivates the use of co-occurrences as features for texture classification. A new algorithm for classifying textures based on co-occurrence feature vectors that are modelled as multinomial density functions is presented.

1. INTRODUCTION

Computer graphics and computer vision have evolved independently for a long time, and there has been very little overlap between these two areas of research. In this paper, a case in which both areas of research are combined will be presented. The key idea is to use a computer graphics method for selecting features which are then employed by a vision algorithm for image interpretation. The domain of image interpretation that is addressed here is the classification of textures.

The selection of suitable features is the most critical part in any classification process. In the area of textural classification, one way to determine if the features chosen are suitable is to synthesize an image which has the given textural features. If the synthetic image resembles the real images closely, then we may safely assume that the features chosen are indeed suitable.

In the first part of this paper, an algorithm for synthesizing textures that obey a set of given co-occurrence matrices is presented, and it is shown that these synthetic images do indeed match their real counterparts very closely. The real images that we were mostly interested in are remotely sensed images of the Landsat/TM sensor as well as ERS-1/AMI data, which generally display textures that are relatively fine-grained.

We claim that for these types of images, co-occurrence matrices are indeed a very good choice. In the second part of this paper, an algorithm for supervised texture classification will be presented. This algorithm differs from existing co-occurrence-based schemes in that it preserves the entire information content of the co-occurrence matrices. There are two basic ideas underlying this algorithm. The first one is to use the multinomial distribution as a parametric model for the co-occurrence features. And the second idea is to use the information theoretic concept of *mutual information* as a function for assessing the weight of evidence.

2. RELATED WORK

Texture segmentation has been one of the key issues in the image processing community and has produced

an abundance of technical papers on the subject. Reviews can be found for instance in [11] or in [20].

Generally, two types of approaches can be distinguished: the structural approach and the statistical approach. The structural approach describes spatial relationships between textural primitives such as blobs or dots. The statistical approach on the other hand describes spatial relationship between individual pixels. These latter methods work well for fine-grained textures, whereas the former approach is mostly used for coarse-grained textures. As our prime interest here is the analysis of satellite imagery that display fine-grained textures, we adhere to the statistical approach. Among the most prominent statistical methods of texture segmentation are various Markov-random field models [3, 4, 9], textural energy filters [15, 16], co-occurrence-based methods [10], and fractal-based approaches [13].

The synthesis of textures has played a vital role in computer graphics [3, 5, 6, 7]. Gagalowicz [5] reports a method for constructing texture fields of order K from a given second order field using the given second order statistic to constrain the higher order field. Since the number of constraints increases exponentially with the size of the field to be generated, this approach is only practical for small image sizes. The textural model used by Gagalowicz is similar to the model used in this paper. However, the method for synthesizing the textures is very different.

Our method is in fact very similar to the approach presented by Cross and Jain [3], who use a Metropolis-type algorithm for producing Markov random field textures, the difference being that Cross and Jain use a different textural model.

The use of synthesis of textures for motivating the choice of textural features is not as widespread, but there are also some references in the literature [2, 14]. Khotanzad and Kashyap [14] use estimates of parameters of stochastic random field models for synthesizing textures that resemble given Brodatz-type textures. Their motivation for using texture synthesis is very

similar to the one underlying this paper: normally, the selection of features is performed by running several classification experiments. If the results are unsatisfactory, then a new set of features is tested. Obviously, this process is very tedious and time-consuming, and also not very robust. Some set of features may work well on one image and not so well on another image. The selection of features on the basis of how well they synthesize to patterns that are visually similar to real patterns is certainly more reliable insofar as it presents a much harder criterion to fulfill.

The approach described here differs from [14] insofar as the texture models used here are co-occurrence matrices instead of random field models. The reason why co-occurrence matrices were chosen here are that they are much easier to handle mathematically, and yet they appear to suffice for the task at hand.

3. CO-OCCURRENCE MATRICES AS TEXTURAL FEATURES

Co-occurrence matrices have been widely used for describing textural properties. In the following, we will briefly summarize the most important definitions. For further reference, see, for instance, [10, 11].

Consider pairs of pixels separated by distance d at some angle α . Generally, distances of one pixel and angles of 0° , 45° , 90° and 135° degrees are used. The $(d = 1, \alpha = 0^\circ)$ -pixel pairs are horizontally adjacent, the $(d = 1, \alpha = 90^\circ)$ -pairs are vertically adjacent, the $(d = 1, \alpha = 45^\circ)$ -pairs are right-diagonal neighbours, and the $(d = 1, \alpha = 135^\circ)$ -pairs are left-diagonal neighbours.

Let m denote the number of gray levels. The (d, α) -co-occurrence matrix C is an $m \times m$ matrix, where an entry c_{ij} of C denotes the number of pairs of pixels separated by distance d at angle α , which have gray values i and j .

For instance, the entries c_{ij} of a $(1, 0^\circ)$ -co-occurrence matrix record the number of horizontal co-occurrences of gray values i and j . The entries c_{ij} of a $(1, 90^\circ)$ -co-occurrence matrix record number of vertical co-occurrences of i and j , and so forth. We shall particularly use co-occurrence matrices in small neighbourhoods of $s \times s$ pixels, where typically $5 \leq s \leq 15$.

The following example of a $(d = 1, \alpha = 0^\circ)$ -co-occurrence matrix illustrates the above definitions:

○	●	○		0	1
●	●	●	0	8	2
●	●	●	1	2	0
<i>image</i>			<i>CM</i>		

The co-occurrence matrix itself is generally not used as a feature vector. Instead, it is transformed into some secondary feature that is then employed for classification. A number of such secondary features are discussed in [11, 20].

The transformation into secondary features leads to a significant loss of information. In some cases, this loss of information is desired, as the new features reduce the information contained in the image to just textural properties. In this paper, we will argue that it

is both possible and desirable to retain the complete information content of the co-occurrence matrices.

Specifically, we will claim that the four primary co-occurrence matrices (horizontal, vertical, left- and right-diagonal) contain sufficient information to synthesize textures that very closely resemble textures from remotely sensed images of the Landsat/TM and ERS-1/AMI sensors.

4. CO-OCCURRENCE-BASED SYNTHESIS OF TEXTURES

In the following, a new method for generating artificial textures based on co-occurrences will be described. The method is a variation of the Metropolis-Algorithm, which has been used in the past for generating Gibbs random fields [3, 9].

Let $c_{i,j}$ be a co-occurrence matrix, whose indices $0 \leq i, j \leq m$ denote the grey levels ranging from 0 to m . The histogram h_i can be easily computed from $c_{i,j}$ by:

$$h_i = \sum_{j=0}^{m-1} c_{i,j}.$$

The first step of the algorithm consists of randomly generating an initial image X_0 that has the desired histogram computed in the above manner. The algorithm now produces a chain of images X_i , $i = 0, 1, \dots$, such that the initial image is iteratively transformed into a final image which has the desired co-occurrence properties of $c_{i,j}$.

We begin by computing the co-occurrences of the initial image X_0 , and determine the error, that is, the distance of the current co-occurrences from the desired co-occurrences. In each subsequent iteration, an image X_{i+1} is derived from X_i , in which the grey values of two randomly selected sites s_1 and s_2 are interchanged provided the new image constitutes an improvement. Thus, the state X_{i+1} differs from X_i in at most the grey values at those two sites. Occasionally, the interchange is performed even if there is no improvement. The probability p with which this happens follows an annealing schedule such that:

$$p = \frac{1}{1 + e^{\Delta/T}},$$

where Δ is the current error and T is a temperature that is slowly cooled down.

The algorithm stops if the variable STABLE is true. The variable STABLE is defined as follows. Let M be the number of pixels in the image lattice. M attempted switches are considered to constitute one iteration. The variable STABLE is true if the number of successful attempts at switching falls below some predetermined percentage of M . Typically, STABLE is set to a value of one percent or less. The algorithm is diagrammed in Fig. 1.

Note that the algorithm can be easily extended to work for several co-occurrence matrices by incorporating them into the error measure:

```

compute histogram from  $c^{desired}$ 
randomly generate an initial image with that histogram
compute initial co-occurrences  $c^{current}$  of that image
set  $\Delta_{old} \leftarrow \|c^{desired} - c^{current}\|$ 

while not STABLE do :
    randomly select sites  $s_1$  and  $s_2$ 
    compute  $c^{current}$  with  $s_1$  and  $s_2$  interchanged
     $\Delta_{new} \leftarrow \|c^{desired} - c^{current}\|$ 
    if  $\Delta_{new} < \Delta_{old}$ 
        switch  $s_1$  and  $s_2$ 
    else
        switch  $s_1$  and  $s_2$  with probability  $u = 1/(1 + e^{\Delta_{new}/T})$ 
    if switched:  $\Delta_{old} \leftarrow \Delta_{new}$ 

```

Fig. 1. An algorithm for generating textures.

$$\sum_k \|c_k^{desired} - c_k^{current}\|,$$

where k ranges over all matrices. Note that the algorithm converges only if a solution exists, that is, matrices are consistent with each other. Also, note that an interchange of two pixels does not require a recomputation of the entire co-occurrence matrix. Instead, it suffices to update the previous co-occurrence matrix in just four positions. A more detailed description of the algorithm can be found in [18].

4.1. Experiments

The experiments were conducted in the following manner: firstly, a training area was marked in an image. Then, the four primary co-occurrence matrices were computed within that area. Finally, the algorithm described above was used to generate an artificial image that has the same co-occurrence properties as the training area. The results are shown in Figs. 2 and 3.

Fig. 2 depicts band 4 of a 512×512 Landsat/TM image showing an area of the tropical rain forest in Brazil. Note that the mixtures of various vegetation types produce highly textured images. The synthetic images are the 64×64 squares superimposed on the original image. The corresponding training areas are the round-shaped areas to the left of the squares.

Fig. 3 shows the same site, this time recorded by the active microwave sensor (AMI) aboard the European satellite ERS-1. Again, the generated textures are the squares, and the corresponding training areas are round-shaped areas to their left. Note the close resemblance of the synthetically generated images to their corresponding training sites (satellite data provided by DLR/DFD, Oberpfaffenhofen, Germany).

5. CO-OCCURRENCE-BASED ANALYSIS OF TEXTURES

From the fact that the synthetic textures agree so well with their real counterparts, we may conclude that the four primary co-occurrence matrices consti-

tute suitable features for characterizing images of the above type.

Unfortunately, the corresponding feature space is of extremely high dimension. Currently known methods usually fail when confronted with such problems. In the past, this problem was usually addressed by extracting secondary features from these matrices. However, the transformation into secondary features leads to a significant loss of information. In the following, we will propose a method that preserves the complete information content of these matrices. The key idea is to use the multinomial distribution as a parametric model for the co-occurrence features.

5.1. The multinomial density function

Suppose we randomly select a sample of n pairs of pixels that are (d, α) -adjacent. Further suppose that the image from which this sample is taken has a normalized co-occurrence matrix p_{ij} . We want to compute the probability of selecting u_{ij} number of pairs having gray values (i, j) or (j, i) .

Note that co-occurrence matrices are special kinds of histograms. The probability of selecting u_{ij} number of pairs can therefore be modelled by a multinomial density function:

$$p(u) = n! \prod_{i,j} \frac{p_{ij}^{u_{ij}}}{u_{ij}!},$$

where,

$$\sum_{i,j} p_{ij} = 1 \text{ and } \sum_{i,j} u_{ij} = n.$$

The term $p(u)$ denotes the probability of encountering u_{ij} number of co-occurrences of gray values (i, j) in a sample of n pixel pairs, where the population from which the sample was taken has a co-occurrence matrix p_{ij} .



Fig. 2. Landsat/TM image with synthetic textures superimposed.

5.2. Mutual information

Once an appropriate feature space and a suitable parametric model has been identified, we can set up a classification scheme. In the following, we will introduce a scheme that is derived from a concept in infor-

mation theory. Its validity and its relation to Bayesian classifiers will be discussed.

Let X be some feature space, and let Ω be the set of object classes to be identified. Let $p(\omega)$ denote the prior probability of class ω . And let $p(\omega|x)$ denote

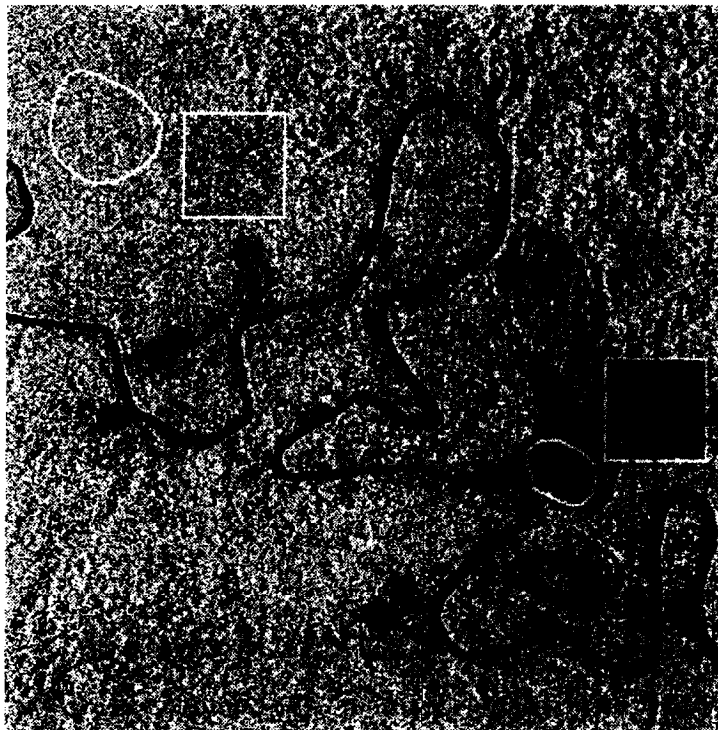


Fig. 3. ERS-1/AMI image with synthetic textures superimposed.

the posterior probability of class ω given the feature vector x . A feature vector x can be regarded as a piece of evidence supporting or refuting some classification hypothesis ω . We want a quantitative measure of the strength of that evidence, that is, of how much the occurrence of some particular $x \in X$ changes the prior probability of some $\omega \in \Omega$. The ratio $p(\omega|x)/p(\omega)$ is a measure of how much more or less likely ω becomes in the light of the evidence provided by x . The logarithm of this ratio,

$$I(x, \omega) = \log \frac{p(\omega|x)}{p(\omega)},$$

is a well-known concept in information theory called *mutual information* (cf. [8]). In our context, $I(x, \omega)$ can be viewed as the strength of the evidence provided by some feature vector x in favor of the classification hypothesis ω . If $I(x, \omega) > 0$, then x supports the hypothesis ω . If $I(x, \omega) < 0$, then x refutes ω , and if $I(x, \omega) = 0$, then x provides no evidence at all (see also [17]).

The function

$$I(x, \omega) = \log \frac{p(\omega|x)}{p(\omega)}$$

seems to require the computation of the terms $p(\omega|x)$ and $p(\omega)$. Both terms are generally not known. Fortunately however, the following holds:

$$\begin{aligned} I(x, \omega) &= \log \frac{p(\omega|x)}{p(\omega)} = \log \frac{p(\omega, x)}{p(\omega)p(x)} \\ &= \log \frac{p(x|\omega)}{p(x)} = I(\omega, x). \end{aligned}$$

In other words, if some approximations $p(x|\omega)$ and $p(x)$ are known, then $I(x, \omega)$ can be easily computed.

Note that the Bayesian classification rule

$$p(\omega_0|x) = \frac{p(x|\omega_0)p(\omega_0)}{\sum_i p(x|\omega_i)p(\omega_i)} = \frac{p(x|\omega_0)p(\omega_0)}{p(x)}$$

is very similar to the above defined weight function $I(\omega, x)$. In fact, the only difference lies in the use of the prior probabilities $p(\omega_i)$. Usually however, these prior probabilities are not known. In practice, they are therefore assumed to be equal, that is, $p(\omega_i) = 1/n$, where n is the number of classes. In this case, the two classification schemes are identical.

In the previous section, we have already identified the multinomial distribution as a suitable parametric model for modelling the co-occurrence feature vectors. We can now use it for approximating $p(x|\omega)$ and $p(x)$. Let p_{ij} be the co-occurrence of the entire image, and let q_{ij} be the co-occurrence within object class ω . Then,

$$p(x|\omega) = n! \prod_{i,j} \frac{q_{ij}^{x_{ij}}}{x_{ij}!},$$

and

$$p(x) = n! \prod_{i,j} \frac{p_{ij}^{x_{ij}}}{x_{ij}!}.$$

Some simple arithmetic yields:

$$\log \frac{p(x|\omega)}{p(x)} = \sum_{i,j} x_{ij} (\log q_{ij} - \log p_{ij}).$$

This term can be easily computed, provided of course that $p_{ij}, q_{ij} > 0$. This can be achieved by smoothing the p_{ij} and q_{ij} , so that indeed $p_{ij}, q_{ij} > 0$, for all i, j .

5.3. Addition of weights

In general, more than one co-occurrence matrix must be used in a classification problem. Therefore, we need a mechanism for combining the evidence provided by several feature spaces.

Let x and y be two feature vectors stemming from two different feature spaces. For instance, let x denote a horizontal co-occurrence matrix, and let y denote a vertical co-occurrence matrix. Assuming independence between the two feature spaces, and also assuming conditional independence with respect to ω , we have:

$$\begin{aligned} I(x, y, \omega) &= \log \frac{p(x, y|\omega)}{p(x, y)} \\ &= \log \frac{p(x|\omega)p(y|\omega)}{p(x)p(y)} \\ &= \log \frac{p(x|\omega)}{p(x)} + \log \frac{p(y|\omega)}{p(y)} \\ &= I(x, \omega) + I(y, \omega) \end{aligned}$$

Under the above independence assumptions, it therefore suffices to merely add the mutual information weights in order to combine the evidence from different sources.

Unfortunately however, these independence assumptions are unrealistic for the following reasons. First of all, conditional and marginal independence usually do not co-occur except in very rare cases. Secondly, the feature spaces that we are interested in, namely co-occurrences of gray levels at varying angles of the same image, are certainly not independent. So in most cases, neither conditional nor marginal independence holds.

In practice however, the error incurred by falsely assuming independence appears to be tolerable. The addition of weights leads to some overestimation of the strength of the evidence. But this effect seems to hit all object classes roughly to the same degree, so

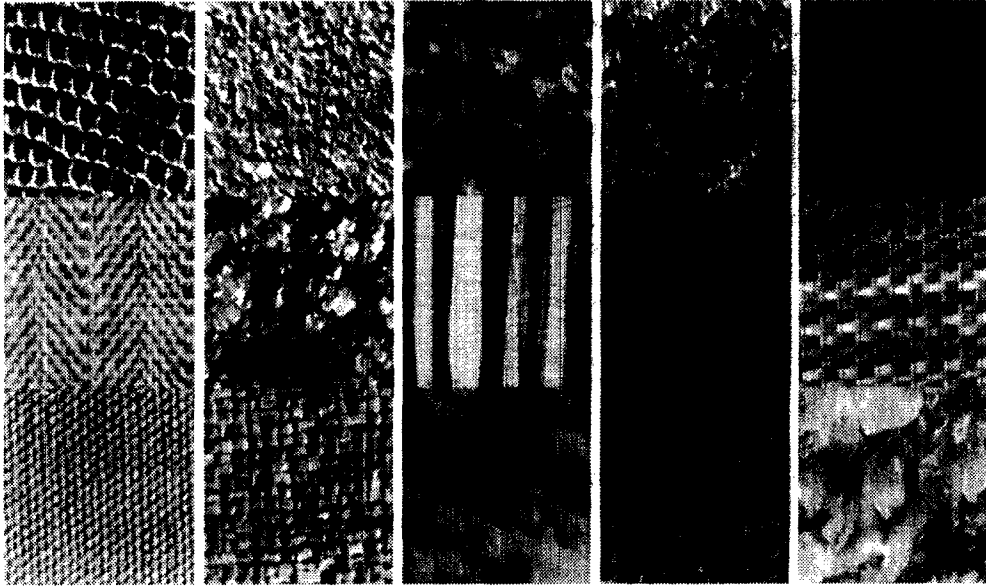


Fig. 4. Brodatz textures.

that the overall effect does not dramatically diminish the classification accuracy.

5.4. The algorithm

The above considerations lead to a new algorithm for supervised texture classification (see also [19]).

Training phase

1. Compute co-occurrences of the entire image.
2. For all object classes, select training areas and compute their co-occurrences.

Classification phase

For all pixels in the image:

1. Compute the k co-occurrence matrices c_{ij}^k in an $x \times s$ neighbourhood of the current pixel; call it x_k .
2. For all object classes ω , compute the weight $\sum_k \log p(x_k|\omega)/p(x_k)$.
3. Determine object class ω_{\max} with the largest weight.
4. If weight is larger than some threshold, then label the current pixel as ω_{\max} ; else label it as *NIL*.

5.5. Experiments

This algorithm was implemented and tested on several different sets of images. The results were very promising.

As a standard test, the algorithm was exposed to a set of 15 Brodatz textures (see Fig. 4) of size 128×128 each. The top half of each image was used as a training pattern, and some smaller patch to be classified was taken randomly from the lower half of each image.

This experiment provided several insights. Firstly, the classification accuracy depends primarily on the number of distinct grey levels that were used to build the co-occurrence matrix. Sixteen grey levels were

required to produce 100% accuracy. The results became totally unreliable if less than 12 grey levels were used. Secondly, the size of the patches played no important role. Even patches of size 16×16 were classified correctly, provided a sufficiently large number of grey levels was used.

The results are summarized in Table 1, where *nbin* denotes the number of grey levels used in co-occurrence matrix, and *errors* denotes the number of misclassified patterns.

5.5.1. *Satellite images.* As a more realistic example, classification tests were done of the ERS1/AMI image introduced earlier. The classes to be identified were "clearcuts (A)", "forest (B)," and "bodies of water (C)." The clearcuts are the darker areas, which are mostly close to the river. The input image with the training areas marked as white boxes is shown in Fig. 5a. The classification result is shown in Fig. 5b. The six grey levels were used to compile the co-occurrence matrices, and the window size (i.e., the size of the neighbourhoods in which the co-occurrence matrices were computed) was 11×11 Pixels.

Table 1. Classification results of Brodatz textures.

<i>pattern size</i>	<i>nbins</i>	<i>errors</i>
64×64	32	0
64×64	16	0
64×64	12	1
64×64	8	5
32×32	16	0
32×32	12	1
32×32	8	5
16×16	16	0
16×16	12	1
16×16	8	3

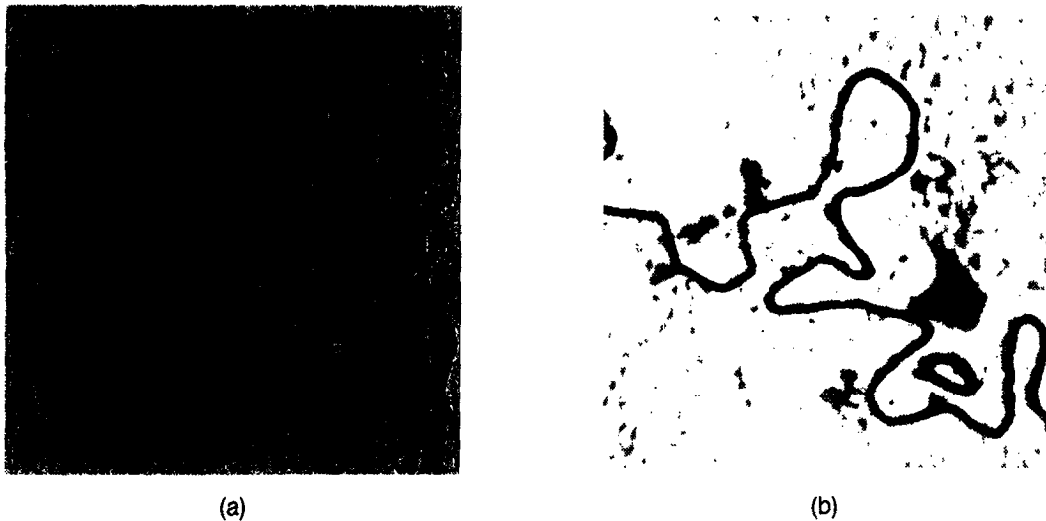


Fig. 5. (a) ERS1/AMI image; (b) classification result.

Some slight modification to the original algorithm was used here: firstly, the moving window, in which the co-occurrence matrices were computed, was rounded off at the corners.

The second modification became necessary, as the river class in the example is not textured at all, that is, its co-occurrence matrices have very few non-zero entries. Such classes cannot be modelled reliably by the above approach, as an entry q_{ij} of value zero needs to be adjusted so that the logarithm $\log q_{ij}$ is defined. While smoothing of the co-occurrence entries eliminates this problem for matrices which have just a few zero entries, it does not produce good results for matrices which consist almost entirely of zero values. Therefore, the following procedure was adopted: if some co-occurrence matrix contains an entry q_{ij} of value larger than 0.8, then this matrix is assumed not to represent a texture class, and membership in that class is established if the corresponding number x_{ij} of co-occurrences in the current image window is larger than some predefined threshold.

To allow some preliminary verification of this result, a visual comparison to a map published by Hönsch [12, p. 194] was done. Hönsch's map was derived from Landsat/TM images and was at least partially verified by ground truth. Visual inspection shows that the result shown here agrees quite well with the earlier map. Some misclassifications occur in the hilly terrain in the upper right part of the image. The shadows cast by the hills produce textures that are very similar to the ones produced by clearcuts. A more thorough investigation is underway.

6. CONCLUSIONS

An algorithm for generating synthetic textures based on co-occurrence matrices was presented. This algorithm was used to imitate real textures taken from satellite images. Surprisingly, these synthetic images resemble their real models so well that they are almost indistinguishable by the human eye. This fact implies

that the co-occurrence features are very well suited for characterizing these types of images.

The consequence of these results is of course to devise an algorithm that uses co-occurrence matrices as feature vectors. Existing co-occurrence-based classifiers usually reduce the information content of the co-occurrence matrices by extracting secondary features. However, those features usually do no longer contain the original grey level and histogram information. In our application domain, this information is vital and should not be eliminated. The algorithm for texture segmentation presented here therefore retains the complete information content. The experiments so far show very good results. Further tests are underway.

Comparisons to other methods of texture segmentation lead to the following conclusions: Markov random field methods are perhaps closest in spirit to the approach described here. However, MRF methods present a number of problems: parameter estimation by Besag's coding method (see [1]) is an optimization process plagued by the usual problems inherent in multi-dimensional search. In comparison, the estimation of parameters for the method described in this paper is trivial, as only co-occurrences need to be computed. Another problem with MRF methods concerns the choice of an appropriate MRF model. There are a variety of possible choices and parameterizations that are not at all obvious. The method presented here has very few parameters for which default values usually work well. Other approaches to texture segmentation such as Laws filters are not suitable for image synthesis.

Future work will be directed towards further improvement of the classification scheme. For instance, a large window size led to inaccurate results at the borders of adjacent texture classes. One way to overcome this problem is to post-process the result by some region-growing method.

Acknowledgements—I would like to thank Mr. Robert Eri for writing the C-code of the synthesis algorithm, and Dr.

Helmut Hönsch for giving technical advice on the geographics of the Brazilian rain forest.

REFERENCES

1. J. E. Besag, Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society Ser. B*, Vol. **36**, 192–236 (1974).
2. R. Chellappa, S. Chatterjee, and R. Bagdazian, Texture synthesis and compression using Gaussian-Markov random field. *IEEE Transactions on Systems, Man, and Cybernetics SMC-15*(2), 298–303 (March/April 1985).
3. G. R. Cross and A. K. Jain, Markov random field texture models. *IEEE-PAMI-5* 25–39 (January 1983).
4. H. Derin and H. Elliott, Modelling and segmentation of noisy and textured images using Gibbs random fields. *IEEE Trans. on Pattern Analysis and Machine Intelligence PAMI-9*(1), 39–55 (January 1987).
5. A. Gagalowicz, A new method for texture fields synthesis: Some applications to the study of human vision. *IEEE Trans. on Pattern Analysis and Machine Intelligence PAMI-3*(5), 520–533 (September 1981).
6. A. Gagalowicz and S. De Ma, Sequential synthesis of natural textures. *Computer Vision, Graphics, and Image Processing* **30**, 289–315 (1985).
7. A. Gagalowicz and S. De Ma, Model driven synthesis of natural textures for 3-D scenes. *Computers and Graphics* **10**(2), 161–170 (1986).
8. R. G. Gallager, *Information theory and reliable communication*, John Wiley, New York (1968).
9. S. Geman and D. Geman, Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images. *IEEE-PAMI-6*, 721–741 (November 1984).
10. R. M. Haralick, Statistical and structural approaches to texture. *Proc. IEEE-67*, 786–804 (1979).
11. R. M. Haralick and L. S. Shapiro, *Computer vision* (Vol. 1) Addison Wesley, Reading, MA (1992).
12. H. Hönsch, Erfassung und Klassifizierung von tropischen Regenwaldformationen und anthropogenen Nutzungen anhand von multisensoralen Satellitenbilddaten am Beispiel Brasilien, Doctoral Thesis, Universität Trier, DLR-Forschungsbericht DLR-FB 93-15, (1993).
13. J. M. Keller and S. Chen, Texture description and segmentation through fractal geometry. *Computer Vision, Graphics, and Image Processing* **45**, 150–166 (1989).
14. A. Khotanzad and R. L. Kashyap, Feature selection for texture recognition based on image synthesis. *IEEE Trans. on Systems, Man and Cybernetics SMC-17*(6), (November/December 1987).
15. K. I. Laws, Rapid texture identification. *Proc. SPIE Conf. on Image Processing for Missile Guidance*, San Diego, 376–380 (1980).
16. K. I. Laws, Goal-directed textured-image segmentation. *Applications of AI II, SPIE* **548**, 19–26 (1985).
17. G. Lohmann, An evidential reasoning approach to the classification of satellite images. European Conf. ES-QAU, Marseille, France, 1991, Proceedings, *Lecture Notes in Computer Science* **548**, Springer Verlag (1991).
18. G. Lohmann. Assessment of textural features for remote sensing applications. *ISPRS Symposium on Spatial Information from Digital Photogrammetry and Computer Vision*, München, Sept. 1994.
19. G. Lohmann. Co-occurrence-based analysis and synthesis of textures. *Intern. Conf. on Pattern Recognition*, Jerusalem, Oct. 1994.
20. P. P. Ohanian and R. C. Dubes, Performance evaluation for four classes of textural features. *Pattern Recognition* **25**(8), 819–833 (1992).