9 Modeling of evolving textures using granulometries

A.J. Gray[(1)], S. Marshall[(2)] and J. McKenzie[(2)]

9.1. Introduction

This chapter describes a statistical approach to classification of dynamic texture images, called parallel evolution functions (PEFs). Traditional classification methods predict texture class membership using comparisons with a finite set of predefined texture classes and identify the closest class. However, where texture images arise from a dynamic texture evolving over time, estimation of a time state in a continuous evolutionary process is required instead. The PEF approach does this using regression modeling techniques to predict time state. It is a flexible approach which may be based on any suitable image features. Many textures are well suited to a morphological analysis and the PEF approach uses image texture features derived from a granulometric analysis of the image.

The method is illustrated using both simulated images of Boolean processes and real images of corrosion. The PEF approach has particular advantages for training sets containing limited numbers of observations, which is the case in many real world industrial inspection scenarios and for which other methods can fail or perform badly.

9.2. Textures and Texture Analysis

It is natural to describe textures qualitatively, in terms of measures such as coarseness, smoothness, granulation, or randomness. However, texture analysis is concerned with the extraction and use of quantitative descriptors of texture.

There are two main categories of textural feature [37], namely *statistical* and *structural* (and some which overlap both categories). Statistical textures include grass, canvas, cork, sand, marble and corrosion. In statistical textures the location of the texture primitive(s) (i.e. meaningful regions) is random and irregular (Figure 9.1). Structural textures consist of repetitions of a basic texture element(s) (texels) or primitive(s). The texels are repeated regularly or with some degree of freedom (Figure 9.2).

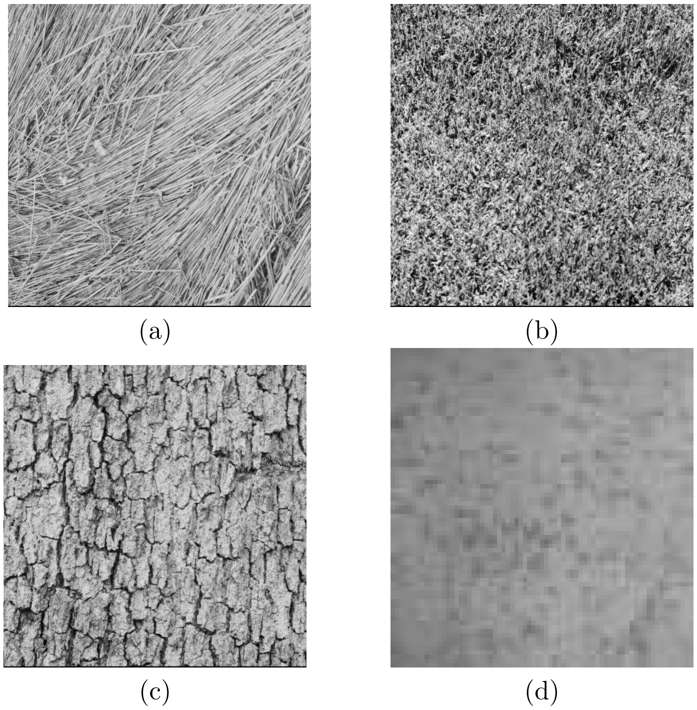

(a) (b) (c) (d)

Figure 9.1. Some examples of statistical textures: (a)–(c) from Brodatz [6]; (d) is an image of corrosion.

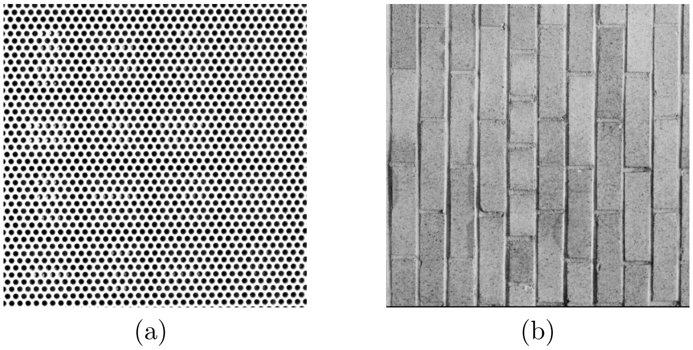

(a) (b)

Figure 9.2. Some examples of structural textures from Brodatz [6].

There are three main areas of texture analysis, namely classification of an unknown texture region to one of a set of texture classes, modeling and describing a given texture region, and segmentation (of distinct textures within an image). Texture classification has been used in many different applications. Some examples are medical image processing, for detecting cancerous cells [52]; automated inspection of machinery, for detection of corrosion [51] and segmentation of landscape features from satellite images [88, 7]. An ideal robust texture classifier should be invariant to changes in rotation, lighting and translation, and be able to detect both very fine and fairly large textures, as well as irregular textures, such as many natural textures, including corrosion (of which rust is a common example).

9.2.1. Approaches to Texture Analysis

We briefly review approaches to the analysis of texture images and the advantages of a morphological approach [60, 81]. There are three main approaches to texture analysis, namely *statistical*, *structural* and *model-based*. Structural methods analyze the texture primitives present in an image and characterize regular textures in terms of arrangements of these texture primitives. The texture primitives can be used as image features for classification purposes. We do not consider structural methods in more detail here, as our main application of interest is corrosion images, which are statistical textures.

9.2.1.1. Statistical Methods

Statistical methods represent the texture by feature variables obtained through measurement. Some of the most common statistical texture analysis techniques [83, 84] are mathematical morphology [20, 79, 80] (Section 9.2.3); spatial gray level co-occurrence or dependence [38], [37, Ch. 9]; gray level run-length distributions [32]; digital transforms [92]; and the autocorrelation function [90, 19].

Spatial gray level co-occurrence (SGLC) matrices [37, 35] characterize texture by the relative frequency of co-occurrence of gray levels of pairs of pixels separated by a given distance and orientation. These are often averaged or summed over different angles. An alternative approach to remove dependence on angle is to compute neighboring gray level dependence matrices [42]. Using regions with certain properties as the primitives instead of pixels has also been suggested. Either the matrices themselves, or more commonly, features computed from them are used for classification.

SGLCs have been very widely used and provide a robust, powerful, and general method for texture analysis, segmentation and classification [38, 32, 16, 95, 88]. For a wide class of images, SGLCs capture much of the texture data, such as degree of coarseness and directionality. Vickers and Modestino [95] achieved 95% accurate classification of various texture types. However, co-occurrence matrices do not contain any shape or size information about texture grains. Some texture features derived from the SGLC [38], including angular second order moment, entropy and correlation measures, do have the advantage of invariance to monotonic gray level transformation. Gray level difference vectors [92, 66] are used similarly to SGLC matrices, in that they compute relative frequency of absolute difference in gray level between pairs of pixels at a given distance for a given orientation, and similar summary measures are used as texture features. Gray level run-length distributions (GLRDs) represent frequency of runs of consecutive pixels with a given gray level in a specified direction [92]. Coarse textures will have a high percentage of longer runs, while fine textures will have more short runs. Galloway [32] achieved 100% classification accuracy using GLRDs for aerial terrain images. Xiaoou [101] found that GLRDs (98-100% accurate) were slightly better than SGLCs on Brodatz [6] and VisTex images (MIT Media Library) [96].

An alternative approach is to view texture in the frequency domain. The Fourier spectrum [92] is found from the discrete Fourier transform of the image. Texture coarseness and directionality are indicated by peaks in 1-D summations of the spectrum or the power spectrum, expressed in polar co-ordinates [4]. Very large images are required to obtain stable values, and local properties cannot be represented, therefore this is unsuitable for texture segmentation or detection of small local areas of texture. Wavelets (Section 9.2.1.2), in contrast, represent both spatial and frequency information. Also, the Fourier transform is not invariant to monotonic gray level transformations, in common with most spatial frequency approaches [40, 97]. Conners and Harlow [16] found the SGLC to perform better than digital transforms. The auto-correlation function (ACF) is the Fourier transform of the power spectrum (spectral density function) [37]. It compares an image with a translated version of itself, thus detecting periodicities in the original image as peaks in the ACF [90]. The ACF is robust to noise and has been widely used in feature recognition. For example, Kurita and Otsu [50] found that it classified Brodatz textures with over 90% accuracy.

9.2.1.2. Texture Models

Textures can be synthesized from a texture model [44, 53, 100], or the parameters of a texture model can be used to characterize an observed texture and enable segmentation [75] or classification. Modeling approaches include random fields [47, 103], especially the Markov random field (MRF) [47], and the autoregression (AR) model [9], as well as wavelets [103, 40], and filter-based representations [71, 72, 67], including Gabor filters [43, 45].

MRFs and AR models both generalize the time-series auto-regressive moving average models [103] to 2-D. Image models are verified as suitable to represent the textures of interest using the training set of texture classes. The model parameter values are chosen to give the best model fit to each observed texture. Texture classification then compares model parameters for an observed image with those for a given texture class. The AR model

[92] specifies pixel gray level as a linear combination of the gray levels in a neighborhood of specified dimensions. The coefficients of this linear combination characterize the texture. Performance depends on neighborhood size, which can be chosen to minimize prediction error.

Random field models represent the texture images as realizations of a random field with a probability distribution concentrated over the most likely images. The joint probability distribution for the image is specified through the conditional probabilities of pixel gray level (or texture type) for a given pixel given the rest of the image [19]. These are assumed to depend only on gray levels (texture types) of neighboring pixels (the Markov property). A suitable form must be specified for the neighborhood and nature of the probabilities. Commonly the Gibbs' random field model is used, in which the joint probability has an exponential form, or the Gauss-Markov random field [49, 13]. However, parameter estimation to fit the probabilities to the images can be difficult. Such models also typically include only nearest neighbor interactions and do not realistically capture higher order structural information in the images. Compared to other techniques, the MRF can achieve very high classification accuracy (100% for many textures) but for a few textures its performance can be very poor [1].

Wavelet transforms and Gabor filters both provide multi-resolution spatial frequency decompositions of an image. Wavelets [56, 57, 64, 100] are a set of orthogonal basis functions which provide a complete representation of the image at different scales. The coefficients of the basis functions provide the multi-resolution texture features. If the number of coefficients is very big, the dimension of the feature vector is decreased. There are many choices of basis function. The Fourier series expansion is one example, using sine and cosine curves. Wavelets are popular but do not always perform well for texture analysis [94, 69, 101]. Results in [94] suggest that the discrete wavelet approach (99% accurate overall) should perform better than most traditional single resolution approaches such as SGLC.

Gabor filters [31, 43, 39] are also widely used. The Gabor functions are used in a similar way to wavelets, as basis sets, though their non-orthogonality makes computation of the coefficients non-trivial. The image is represented as a linear superposition of Gaussian modulated complex sinusoids (whereas the Fourier representation [76] uses only sinusoids). Various studies have compared the Gabor filter with other texture classifiers and segmentation schemes, mainly MRF [39] variations and wavelet methods [68]. Haley and Manjunath [39] found that the Gabor filter outperformed MRF methods, while Manjumath and Ma [58] found that the Gabor filter and multi-resolution simultaneous AR models outperformed wavelets. However it did not perform well in a comparison of filtering methods by Randen and Husoy [72]. Multichannel filters based on the Gabor filter, but with much lower computational cost, can perform almost as well [71]. Gabor filtering has various desirable properties over other multi-resolution methods [103, 59] and the advantage of strong parallels with the low level visual processing of humans and other mammals.

Finally, texture analysis of color images [28] is a higher dimensional problem than for gray level textures. Transforming an image from RGB space to a different color space, such as LUV space [82], can be useful for producing a more perceptually consistent image segmentation [62] when combined with morphological operators. Köppen et al. [48] proposed a generalization of gray-scale morphology to color images. Monadjemi et al. [63] proposed Walsh transforms for color texture classification, while Chindaro et al. [13] used color space fusion for classifier combination.

9.2.2. Evolving Textures

For evolving textures, such as occur in a corrosion process where a few small 'spots' gradually increase in size and number until they cover the surface of the material (see Figure 9.8), or in diseased tissues imaged over time, the classification problem is one of relating a given sample of texture to a specific time within a texture evolution process rather than to one of a series of unconnected texture classes.

Most conventional texture classifiers compare characteristics of the texture with those of a known class and place it to the nearest class, according to some measure of closeness, e.g. the maximum likelihood (ML) or Euclidean distance criteria. For example, Liu et al. [52] use a quadratic discriminant function; Kyvelidis et al. [51] use fuzzy logic to classify defects (the results are used on a series of corrosion images to examine the growth process); Garcia-Sevilla and Petrou [33] use the nearest-neighbour rule with various distance measures; Clausi [14] uses a 2-class Fisher linear discriminant rule to project an n-dimensional class feature set to a one-dimensional vector providing an optimal separation of 2 classes, and a series of ML classifications is then used to assign an observation to the class with the highest number of hits; and Rajesh et al. [70] use unsupervised classification based on the ISODATA algorithm [29], which minimizes a least squares objective function.

The PEF method differs in that, instead of using a series of comparisons with predefined classes, it computes a set of parameters directly related to the class or state of evolution on a continuous scale, from the statistical texture characteristics. The removal of the direct comparison stage enables the classification of images that are not from any of the training classes, but are from classes intermediate to these, as long as the growth pattern is sufficiently well represented by the training examples. More importantly, the use of all instances of the texture over time to train the classifier means that for smaller training sets there is more information available for training. Chantler and Stoner [8] have shown that the use of combined sequences of data measurements can greatly decrease the error rate of feature classification over methods using each measurement in isolation.

9.2.3. Advantages of Morphological Analysis

Series of evolving texture images often consist of images which initially contain very little detail, then the texture builds up through time by the placement of more and/or larger objects in the image. This is the case with images of corroding metal, for example, considered in Section 9.5, in which the metal begins as a smooth surface, then becomes damaged and a small seed or grain of corrosion appears. Further grains may appear at other positions in the image and these grains increase and enlarge over time to cover the image plane. Alternatively an evolving texture may coarsen over time, as in images of soil becoming wetter. In either case such images are well suited to a morphological analysis, which extracts size and shape information from texture elements present in the image at a given time.

Mathematical morphology [60, 81] is a methodology for shape-based non-linear image filtering using set operations. It has a very broad spectrum of applications and can be applied to localized textures, pixel by pixel [26, 27], particularly where a texture is composed of well-defined grains or where variation in the grains is important. This approach allows for good discrimination between the texture of interest and any background clutter which does not share similar shape properties. In particular, although the granulometry (Section 9.3) is a statistical method, it can be related to the 3-D physical properties of a material and its surface texture. While covariance-based methods, such as autoregression, difference statistics, Fourier transforms and SGLC, only describe the texture process up to second-order characteristics, morphological methods can capture higher order properties of the spatial random processes [3] and no other approach directly uses shape information in the image. Morphological tools can also be directly translated to Boolean algebra, and therefore are easily implemented in hardware for efficient integration into digital systems for faster processing [2].

9.3. Granulometries

Granulometries are well known morphological tools [20, 79] which consist of an iterative sequence of morphological operations. The simplest granulometry consists of a series of structural openings [60, 20], which we use here.

A morphological opening, SoB, of an image S by a structuring element (SE) B, is the image consisting of the union of every translation x of B that is totally contained within the image set in S (or which fits beneath the image surface in the gray level case), giving a coarse representation of the original image:

$$SoB = \cup\{B + x : B + x \subset S\}. \tag{9.1}$$

For gray-scale images, opening with a 2-D SE is faster than using a 3-D SE, however the latter can also extract information on the gradient and contours of the surface, as well as surface shape information.

An opening-granulometry uses a series of SEs of increasing size r, that remove all image particles within which the SE cannot fit. Grains of smaller size are removed from the image by small scale SEs and larger grains are removed sequentially as the SE is increased in size until no image content remains. The volume $\Omega(r)$ of the opened image is recorded at each step, i.e. the sum of all pixel intensity values in the image (or a sum of remaining pixels in the binary case). These volumes are then normalized using the initial volume $\Omega(0)$, to define a cumulative probability distribution function, the *size distribution*,

$$\phi(r) = 1 - \Omega(r)/\Omega(0),\ r = 0, \ldots, R, \tag{9.2}$$

which increases monotonically from 0 to 1 as r increases. This is the proportion of the volume removed up to and including step r. The differences between the normalized values define the *pattern spectrum* of the image (Figure 9.3):

$$\Phi(r) = \phi(r) - \phi(r-1),\ r = 1, \ldots, R, \tag{9.3}$$

i.e. the proportion removed at stage r, which approximates the derivative of $\phi(r)$ and is a probability mass function.

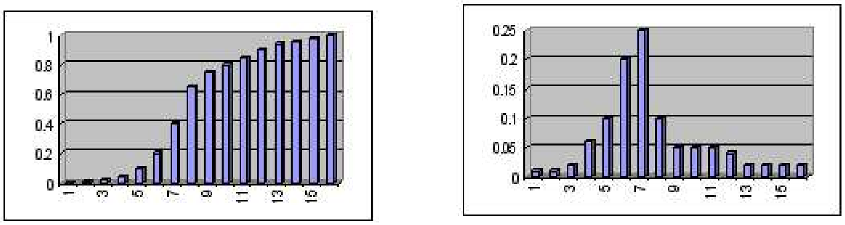
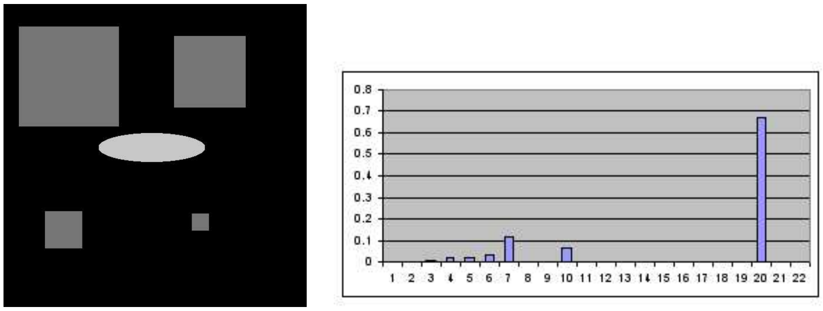


Figure 9.3. A size distribution (left) and its corresponding pattern spectrum (right).

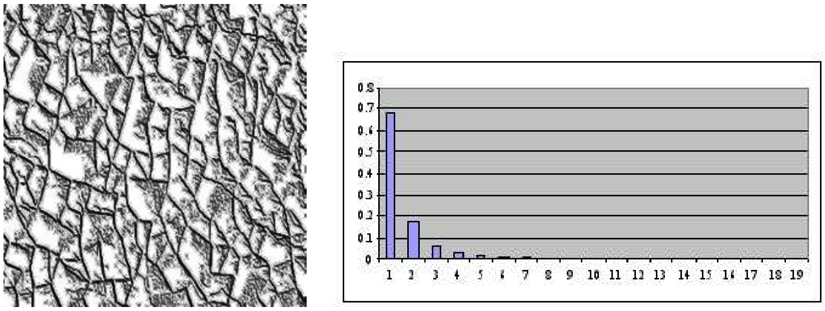
The pattern spectrum profile captures the size and shape information from the foreground texture detail at differing scales. The more 'peaked' the profile, the more closely the shape of the SE fits the texture structure, as sharp drops indicate a large number of grains of a given size and similar shape to the SE. Smooth graphs indicate a more even spread of sizes of grains, or grains which are not similar in shape to that of the SE. Figures

9.4-9.5 show examples for a simple gray-scale case and a real texture image. In Figure 9.4, the image contains 4 squares of different sizes and an ellipse, and the SE is square. The pattern spectrum contains several separate spikes, one for each of the 4 squares (which drop out one after the other as complete objects, as they are the same shape as the SE), and several contributed to by the ellipse (which differs in shape from the SE, and so is gradually eroded to become more rectangular in shape before disappearing completely). Figure 9.5 shows a 3-D (gray-scale) image and the SE used was a 2-D square. There is no clear separation of the grains by the background, and the use of a SE with dissimilar structural shape to that of the grains in the image results in a pattern spectrum which consists of a slowly decreasing curve, rather than a series of spikes corresponding to complete objects disappearing.



(a) (b)

Figure 9.4. (a) Image consisting of differently shaped grains, and (b) its pattern spectrum obtained using a square SE; the horizontal axis represents size r and the vertical axis s the proportion of image volume removed.



(a) (b)

Figure 9.5. (a) A texture image of dried paint[6], and (b) its pattern spectrum from a square SE.

As the pattern spectrum is a probability mass function, its statistical moments can be calculated and used as texture feature descriptors [25, 60, 54]. These are referred to as the *granulometric moments* [20]. The ith order moment is:

$$M_i = E(r^i) = \int_r r\Phi(r)dr \approx \Sigma r^i \Phi(r), \tag{9.4}$$

in which r is a measure of size of the SE.

More general granulometries that do not necessarily use an opening can be defined [22]. In each case there is a parameterized set of operators $\{\psi_r\}$ and the volume of the image operated upon by $\{\psi_r\}$ is computed to form the size distribution. In our application $\{\psi_r\}$ is an opening by a single SE rB, where B is a fixed primitive (the *generator* of the granulometry) and r is a scaling factor. Another frequently used granulometry takes $\{\psi_r\}$ as a union of openings from different SEs $rB_1, rB_2, \ldots, rB_m$ [5].

Use of the Euclidean granulometry [24] ensures that the granulometry acts strictly as a sieve, in which grains or image areas are removed depending only on their size and shape relative to the SE(s). The Euclidean granulometry is a family of operations $\{\psi_r(S) = S \ o \ rB\}$ which is *anti-extensive* (i.e. $\psi_r(S) \subseteq S$); *increasing* ($A \subset B \Longrightarrow \psi_r(A) \subset \psi_r(B)$, i.e. the sequence of processed images has the same order as the original images); *fulfilling the mesh property* ($\psi_r\psi_q = \psi_q\psi_r = \psi_{max(r,q)}, r, q > 0$, i.e. use of several SEs is equivalent to using only the largest); *scale invariant* ($\psi_r(S) = r\psi_1(S/r)$, i.e. scaling S by $1/r$, sieving by ψ_1, then re-scaling by r is the same as sieving S by ψ_r); and *translation invariant* ($\psi_r(S_x) = (\psi_r(S))_x$, where S_x is the image S translated by x) [21]. These five properties limit the type of operators and SEs that can be used [80, 21, 24]. In particular, the SE used for opening must be convex.

Note that opening of the image foreground (the objects) and the background gives complementary information. Opening the foreground provides information on size and shape, while opening the background provides infomation on spatial distribution of the objects in the image. Use of the pattern spectrum moments from both foreground and background is therefore recommended.

It has been shown [77, 86] that the granulometric moments for an image containing disjoint grains are asymptotically normally distributed with increasing numbers of grains. For a limited amount of grain overlap, the distribution can be assumed to be approximately asymptotically normal. This asymptotic normality is one reason why the granulometry has proven to work well for texture classification in conjunction with a Gaussian ML classifier [78, 77]. We use this as a benchmark for the PEF approach.

The theoretical properties of granulometries and their properties in local texture classification are well researched. They have achieved classification accuracy in the high 90% range using granulometric features in a Gaussian ML classifier [26, 11] on Brodatz textures [6], even in the presence of noise and visually similar textures. Other morphological tools have also been used, with a fuzzy hit-or-miss texture transform achieving 96.9%

accuracy for Brodatz textures in [102]. The granulometric approach has proved successful in varied applications, including blood cell analysis and chromosome overlap detection [12]; detecting changes in trabecular bone with an ML classifier [10]; analysis of electrophotographic (toner particle) images [25]; segmentation of digital mammograms [3], and counting white blood cells [91].

Ayala and Domingo [1] proposed new descriptors for binary and grayscale images based on new spatial size distributions, that generalize the usual granulometric size distribution. These combine a granulometric analysis of the image with a comparison between the geometric covariograms for binary images, or, for grayscale images, the ACF (auto-correlation function) of the original image and its granulometric transformation. The new descriptor was better where a finer description of the image was required. The ACF on its own did not perform as well as using granulometry. Ayala and Domingo [1] compared their new method using a Bayesian classifier with some of the most common existing methods, including an opening-granulometry performed only on the foreground (thereby losing shape distribution information, which the new method included). There was no undisputed best classifier, however overall they concluded that the granulometry with added spatial data was at least as good as any of the other methods (MRFs, SGLCs, fractal dimensions [17, 46, 93] and Gabor filters). Performance increased dramatically when spatial distribution data was included with the foreground granulometric moments. Even a foreground granulometry performed at least as well overall as a simple SGLC, for example.

Sivakumar and Goutsias [85] developed morphologically constrained Gibbs' random fields and applied them to texture synthesis and analysis. It was shown that under certain conditions the ML estimators of the morphologically constrained GRF parameters could be approximated using the granulometric pattern spectrum, providing a much easier and faster implementation of GRFs. A computationally simple morphological Bayes' classifier then achieved 98% correct classification on Brodatz textures.

9.4. Parallel Evolution Functions

The Parallel Evolution Function (PEF) approach [54] to time classification of evolving textures is now described. It may be used with any calculable descriptors that characterize the image at different times. Here it makes use of the granulometric pattern spectrum moments as image features. These are calculated, as described above, from a granulometric opening of the texture image.

The PEF approach is a statistical method which places an image from an evolving process directly to a position on the evolution time scale of the process, rather than compare the observed features with all available time classes and allocate to the closest class. It relates the evolution parameters of the model assumed to generate the texture images to the observable granulometric moments over a series of time steps, using multiple linear regression (MLR) of the moments, as the dependent variable, on the evolution parameters, as the explanatory variables. (Note that the moments are *random* functions of the *fixed* parameters of the evolution process). Using a training set of simulated images for which the underlying evolution parameters are known, a multiple regression model is fitted for each chosen moment, giving an "evolution function" relating the (mean over the available observations at each time step of the) moments of the evolving textures to the parameters controlling the evolution. The result is a set of these evolution functions; one for each chosen moment, hence the term "parallel". Once the PEFs are fitted, for any new image, for which the time state is unknown, the granulometric moments of the image are calculated. Each of the PEFs could be used singly to predict the evolution parameters and then the time of evolution, however each of the PEFs involves the same parameters. Therefore the PEFs are combined as described below.

At each of $n+1$ times $t = 0, \ldots, n$, there are l images, giving $l \times (n+1)$ moments of a given order i, $i = 1, \ldots, m$. Let $M_i(t)$ denote the training sample mean (averaged over the l images) of the ith moment at time t for times $t = 0, 1, \ldots, n$, giving an $(n+1)$-vector of moments of order i, $\mathbf{M_i} = (M_i(0), M_i(1), \ldots, M_i(n))^T$. $\mathbf{P_j}$ is the $(n+1)$-vector $\mathbf{P_j} = (P_j(0), P_j(1), \ldots, P_j(n))^T$, where $P_j(t)$ is the jth evolution parameter, $j = 0, \ldots, J$, at time t (used as an explanatory regressor variable). The least-squares MLR model is given by

$$\mathbf{M_i} = \Sigma_{j=0}^{J} b_{ij} P_j + r_i, \; i = 1, \ldots, m, \tag{9.5}$$

i.e.

$$\begin{pmatrix} M_i(0) \\ M_i(1) \\ \vdots \\ M_i(n) \end{pmatrix} = \begin{pmatrix} P_0(0) & P_1(0) & \ldots & P_J(0) \\ P_0(1) & P_1(1) & \ldots & P_J(1) \\ \vdots & \vdots & \ddots & \vdots \\ P_0(n) & P_1(n) & \ldots & P_J(n) \end{pmatrix} \begin{pmatrix} b_{i0} \\ b_{i1} \\ \vdots \\ b_{iJ} \end{pmatrix} + \begin{pmatrix} r_i(0) \\ r_i(1) \\ \vdots \\ r_i(n) \end{pmatrix}, \tag{9.6}$$

where the $(n+1)$-vector $\mathbf{P_0} = (1, 1, \ldots, 1)^T$, and $\mathbf{r_i}$ is an $(n+1) \times 1$ column vector of random errors or disturbance terms representing the deviations of the ith order moments in the vector $\mathbf{M_i}$ from the expected moments given by $\Sigma_{j=0}^{J} b_{ij} \mathbf{P_j}$. The coefficient vector $\mathbf{b_i} = (b_{i0}, b_{i1}, \ldots, b_{iJ})^T$ is estimated by minimising the error sum of squares [61]:

$$\Sigma_{t=0}^{n} [M_i(t) - \Sigma_{j=0}^{J} b_{ij} P_j(t)]^2, \tag{9.7}$$

giving

$$\hat{\mathbf{b_i}} = (\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T\mathbf{M_i}. \tag{9.8}$$

We have such a regression model $\mathbf{M_i} = \mathbf{P}\mathbf{b_i} + \mathbf{r_i}$ for each moment used $(i = 1, \ldots, m)$. The goal is to estimate the evolution parameters from the moments, and having acquired these parameter estimates, to use them to derive the time status of the image. The same evolution parameters are involved in each regression equation, which we use as follows. Let $\mathbf{M}$, $\mathbf{B}$ and $\mathbf{P}$ be defined as:

$$\mathbf{M} = \begin{pmatrix} M_1(0) & M_2(0) & \ldots & M_m(0) \\ M_1(1) & M_2(1) & \ldots & M_m(1) \\ \vdots & \vdots & \ddots & \vdots \\ M_1(n) & M_2(n) & \ldots & M_m(n) \end{pmatrix} = (\mathbf{M_1} \mid \mathbf{M_2} \mid \ldots \mid \mathbf{M_m}), \tag{9.9}$$

$$\mathbf{B} = \begin{pmatrix} b_{10} & b_{20} & \ldots & b_{m0} \\ b_{11} & b_{21} & \ldots & b_{m1} \\ \vdots & \vdots & \ddots & \vdots \\ b_{1J} & b_{2J} & \ldots & b_{mJ} \end{pmatrix} = (\mathbf{b_1} \mid \mathbf{b_2} \mid \ldots \mid \mathbf{b_m}), \tag{9.10}$$

and

$$\mathbf{P} = \begin{pmatrix} P_0(0) & P_1(0) & \ldots & P_J(0) \\ P_0(1) & P_1(1) & \ldots & P_J(1) \\ \vdots & \vdots & \ddots & \vdots \\ P_0(n) & P_1(n) & \ldots & P_J(n) \end{pmatrix} = (\mathbf{P_0} \mid \mathbf{P_1} \mid \ldots \mid \mathbf{P_J}). \tag{9.11}$$

Then the MLR model can be summarized by $\mathbf{M} = \mathbf{PB} + \mathbf{R}$, where $\mathbf{R} = (\mathbf{r_1} \mid \mathbf{r_2} \mid \ldots \mid \mathbf{r_m})$. Then $\hat{\mathbf{M}} = \mathbf{P}\hat{\mathbf{B}}$ where $\hat{\mathbf{B}} = (\hat{\mathbf{b_1}} \mid \hat{\mathbf{b_2}} \mid \ldots \hat{\mathbf{b_m}})$ contains all the least squares estimates of the coefficients from the regressions for each of the moments $(i = 1, \ldots, m)$ separately.

Given a new image, the evolution parameters $\mathbf{P}$ are then estimated from the observed moments as $\hat{\mathbf{P}} = \mathbf{M}\hat{\mathbf{B}}^+$ where $\hat{\mathbf{B}}^+$ is the pseudo-inverse of $\hat{\mathbf{B}}$ and where $\hat{\mathbf{B}}^+ = \hat{\mathbf{B}}^{-1}$ if $\mathbf{B}$ is non-singular [19]. The assumed growth model equations relating evolution parameters to time are used to convert estimated evolution parameters to estimated evolution time (Section 9.6.1). The process is illustrated in Figure 9.6.

9.4.1. Model Fitting Issues

Use of ordinary least squares (OLS) regression, as described, assumes that in each of the m regression models, the errors $r_i(0), \ldots, r_i(n)$ are uncorrelated

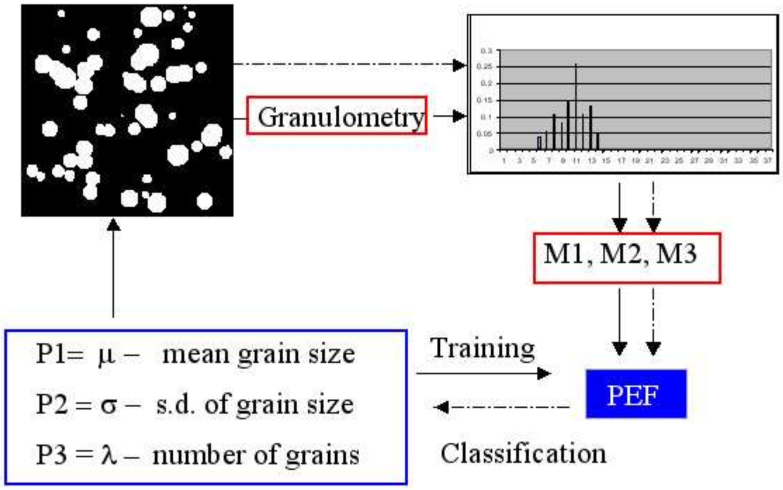


Figure 9.6. The training and classification process for the PEF approach, using parameters P1–P3 and moments M1–M3; solid and dotted lines indicate training and classification steps respectively.

and have the same variance $\tau_i^2, i = 1, \ldots, m$, and the optimal estimators of the regression coefficients are given as in Eq. (9.8). If these assumptions are invalid, these estimated coefficients will be suboptimal. Weighted least squares regression allows for differing error variances within a given regression model, while generalized least squares (GLS) regression allows for correlated errors also, as are likely to be present in data collected through time. GLS assumes that $cov(\mathbf{r_i}) = \tau_{\mathbf{i}}^{\mathbf{2}}\mathbf{W_i}$, $i = 1, \ldots, m$, where $\mathbf{W_i}$ is an arbitrary positive definite $(n+1)\times(n+1)$ variance-covariance matrix, rather than $cov(\mathbf{r_i}) = \tau_{\mathbf{i}}^{\mathbf{2}}\mathbf{I_{n+1}}$, a constant multiple of the $(n+1)\times(n+1)$ identity matrix, and the minimum variance linear unbiased estimators of the regression coefficients are then [99, 73]

$$\hat{\mathbf{b_i}} = (\mathbf{P}^T\mathbf{W_i}^{-1}\mathbf{P})^{-1}\mathbf{P}^T\mathbf{W_i}^{-1}\mathbf{M_i}. \tag{9.12}$$

However, estimating the covariance matrix means that the estimated coefficients $\hat{b}_i$ no longer have minimum variance and may have greater variance than the OLS estimates. Reliable estimation of individual variances and covariances requires large amounts of data and unreliable estimation is likely to lead to estimates with poorer model fit than with OLS. For both the simulated and real images used below, the sample variances of the moments were in general not constant over time, however we use OLS regression modeling, owing to limited availability of data. This may adversely affect model fit.

It will be seen (Section 9.6.1) that in our model all evolution parameters increase with time, and hence there are likely to be collinear relationships between the explanatory variables [73]. The likelihood of the fitting of the regression coefficients becoming unstable (i.e. small changes in the data or explanatory variables leading to large changes in the estimated coefficients) due to collinearity was tested, using eigenanalysis [73] of the covariance matrix of the explanatory variables as well as altering the order of the chosen explanatory variables. Near-zero eigenvalues indicate linear dependencies between the explanatory variables. For both the simulated and real data, collinearity was found to be present, especially for the background data, however this did not appear to cause model instability. Nevertheless, centering the explanatory variables, or using principal components regression [73] may be helpful.

Assessing model fit using statistical tests requires that the moments (or residuals) in the regression model are normally distributed [99, 73]. For the simulated images, normal probability plots and histograms indicated that the distributions of the residuals were reasonably symmetric but heavy tailed. The plots for the background moments of the real images were closest to a normal distribution.

9.5. Application to Corrosion Images

Industrial inspection is a well-established field, essential to the safe and economical running of any business involved in manufacturing or requiring to maintain equipment. Component damage or defects often consists of cracks, discoloration, or variations in size or texture [25, 89, 51, 82]. Mechanical parts, such as aircraft engine components [74], showing signs of damage may be routinely replaced at regular intervals, whereas optimal replacement times may be different. Having the ability to relate degree of damage to weakening of a component is therefore important.

The motivation is to develop a method for automatic detection and classification of corrosion of metal parts according to its seriousness in terms of component weakening or loss of functionality. Figure 9.7 shows samples of turbine blades with varying degrees of corrosion. Clearly it would clearly be desirable to remove blades as corroded as that in D, before they reached this degree of damage.

Figure 9.8 shows 4 images of a mild steel plate, sprayed regularly with a dilute saline solution over a period of 9 days, as it corrodes over time. The resulting series of 10 images (at times $t = 0 - 9$ days) shows the evolution, from no corrosion to almost complete oxidation of the surface of the plate. The plate changed from gray to brown during the experiment, however there was insufficient variation in color for color itself to be useful for prediction of time state. The 10 images of 1400×1400 resolution were converted to

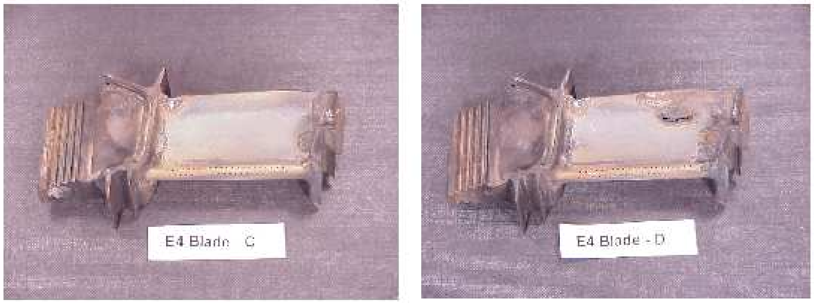
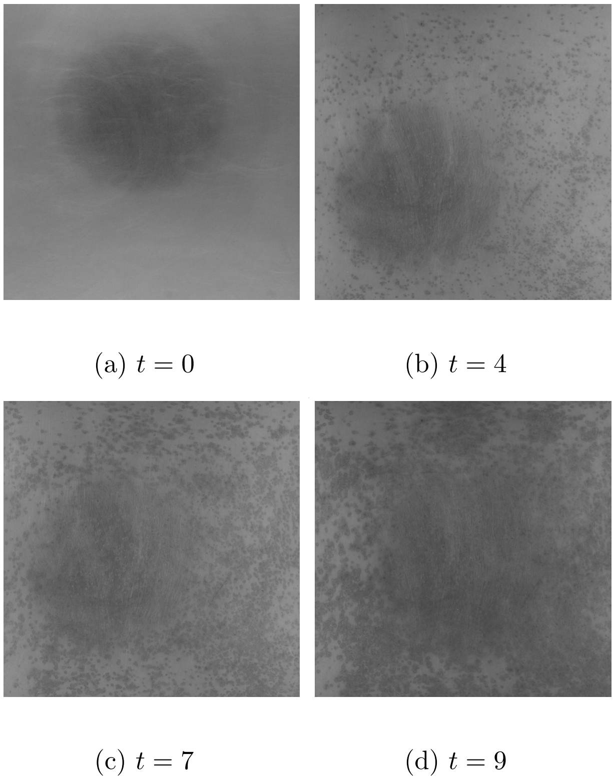


Figure 9.7. Turbine blades showing different degrees of corrosion: some damage visible on C and severe damage on D.



(a) $t = 0$

(b) $t = 4$

(c) $t = 7$

(d) $t = 9$

Figure 9.8. Gray-scale corrosion images at times t = 0, 4, 7 and 9 days.

gray-scale (as shown) for gray-scale granulometry to be used. The blob-like textures observed in these images are well suited to a morphological analysis, as provided by granulometry.

Corrosion appears as an irregular texture, which evolves according to an underlying chemical process controlled by many factors, such as environmental humidity and acidity, initial surface roughness and type of material. The detection of corrosion is seldom easy, as any image of a component will contain shading and shapes that are not the result of corrosion but which may look similar to the effects of corrosion. The PEF method assumes that the evolution of the underlying chemical process results in a parallel evolution of the visual texture, and approximates this relationship in order to classify an observed texture to a precise point on the evolution scale. The texture process is modeled as a time dependent stochastic spatial point pattern described by random variables whose value depends stochastically on parameters which are functions of time. These underlying parameters can be used to quantify component weakening, through mechanical testing or comparison with components that have been assessed by a human expert. However, these parameters describing the evolutionary state of corrosion cannot be observed directly from new corrosion images. The pattern spectrum moments of the corrosion image are therefore used as a means of estimating the underlying hidden parameters. These estimates can then be used as a measure of the state of evolution or severity of corrosion, by finding the equivalent length of time taken, under set conditions, for the corrosion to form and therefore the degree of weakening caused (if this is known for the training data).

9.6. Modeling the Texture

9.6.1. Image Simulation

The methodology is illustrated first on binary, then gray-scale, simulated images. Simulation permits fitting of the regression models relating the underlying model parameters to the observable granulometric moments, as the values of the parameters are known for a given evolution time of the texture. The choice of model and parameters controlling the evolution was based on observations of corrosion forming under natural conditions. This appears as a few small spots, which grow slowly at first, then size, and then also the number of spots increase rapidly, until the entire surface is covered. Therefore the model assumes corrosion to be a spatial point pattern growing from randomly distributed seed points [65].

A Boolean point process model [18, 41, 79], a random process in which random shapes ('grains') are positioned according to an independent point process [36], is used to represent the image textures, although any suitable model can be used. Grains are scaled to random size and placed randomly (according to a Uniform distribution) in the image plane with grains simulated according to a Poisson point process with grain intensity $\lambda(t)$ (number of events per unit area) controlling the number of grains in the image. The Poisson process assumes that the grain positions are disjoint but grains may overlap. The values of the parameters of the model (the intensity and the grain size parameters), the "evolution parameters", are functions of the evolution time t.

The parameters were chosen empirically to give a series of realistic-looking textures, beginning with no coverage of the image background, increasing rapidly to almost complete coverage (at time $t = 20$). Texture growth is achieved by increasing grain intensity $\lambda(t)$ and mean grain size $\mu(t)$ with time. The variance of the grain dimensions also increases with time, to allow for new seed points forming once larger grains are already formed. An increase in the rate of corrosion with time can be explained as being due to the increased surface area exposed by surface roughening [98]. Grain size at time t is taken as a folded Normal random variable (i.e. absolute grain size is used), with mean $\mu(t)$ and standard deviation (s.d.) $\sigma(t)$. The grain intensity (no. of grains) $\lambda(t)$ and size standard deviation $\sigma(t)$ are assumed to increase linearly, while mean grain size $\mu(t)$ increases quadratically with time, as illustrated in Figure 9.9. This results in grains that both grow and overlap to a greater extent as time increases.

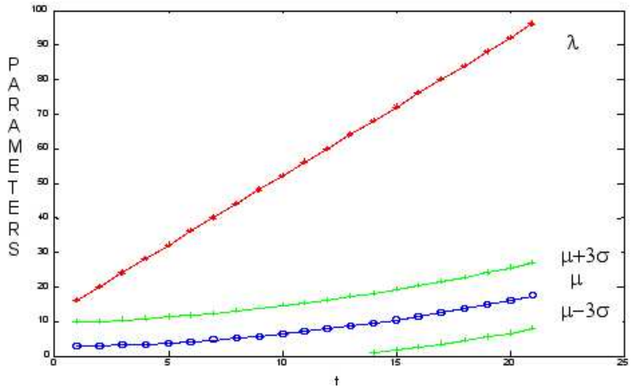
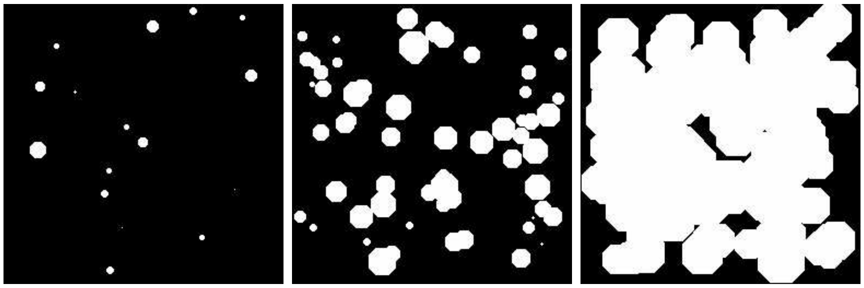


Figure 9.9. Assumed evolution path of the corrosion model parameters, using $\mu(t) = 0.03t^2 + 0.133t + 2.61$; $\sigma(t) = 0.0416t + 2.349$; and $\lambda(t) = 4t + 16$. Note that some small grains are present at time 0.

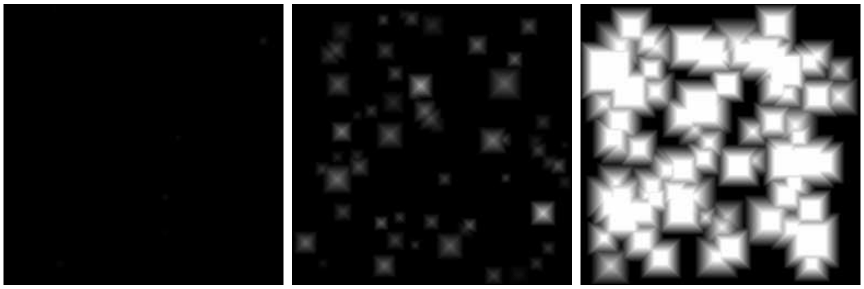
For a new image, once the parameter values are estimated from the moments, each of the equations relating the parameters to time is inverted to predict time t for that image and these various predictions are averaged. Alternatively, a single equation may be chosen on the basis of minimizing classification error for the training images, giving a single predicted time.

Figures 9.10–9.11 show example textures using binary octagonal grains and square-based gray-scale pyramidal grains (clipped to a maximum gray level of 255). These simulated images were generated using new seed points in every instance and not grown from one set over time. Both models use parameter evolution as in Figure 9.9. For the octagons (approximation to a disc) the grain size is the radius. For the pyramids, the mean base length is taken as $2\mu(t) + 1$, for comparability with the octagons, and the mean step size is also set to $\mu(t)$, so that gradient and overall gray level increase with time.



(a) $t = 0$ (b) $t = 11$ (c) $t = 20$

Figure 9.10. Binary octagonal textures at times $t = 0$, $t = 11$ and $t = 20$.



(a) $t = 0$ (b) $t = 11$ (c) $t = 20$

Figure 9.11. Gray-scale pyramid images at times $t = 0$, $t = 11$ and $t = 20$.

9.6.2. PEF Modeling

We use the first three moments of the pattern spectrum, $M_i(t), i = 1, 2, 3$, at each time t (see Section 9.3), so there are three regression equations. We chose an MLR model depending on a subset of the first five powers of the Boolean model parameters, μ, σ, λ, μ^2, σ^2, $\lambda^2, \ldots, \mu^5, \sigma^5, \lambda^5$, which results in a set of 15 candidate explanatory regressor variables $p_1, p_2, \ldots, p_{15}$ (and 16 coefficients if an intercept is included). This was done to increase the chances of good model fit. The "best" three MLR regressor variables of these 15 candidates for each regression were chosen as those giving the lowest classification error on the training images. This is very time consuming, and a stepwise variable selection procedure may be more useful in practice [73]. Note that these were not necessarily those variables giving best fit of the regression models. It was generally found that the best results were obtained by restricting the number of regressor variables and using separate regressions for the foregound and background moments. Three variables were used here so that $\mathbf{B}$ is a square matrix when modeling 3 moments, but more could be used.

The evolution time is estimated as the (nearest integer to the) mean of the estimated times from the foreground and background regressions.

9.6.3. Benchmark Methods

The most commonly used classifier is the Gaussian maximum likelihood (ML) classifier [30], however (both for this and the optimal Bayes' classifier assuming multivariate normality of the feature variables), the number of observations needs to be larger than the number of variables used, for the covariance matrix of the variables to be non-singular [34]. This severely limits the usefulness of these methods in this context, given the very small number of observations often likely to be available in practice. The ML classifier was therefore used with the pooled covariance matrix from all time classes (the MLP method). In some cases MLP performed worse than ML using individual covariance matrices and in other cases it was better, but it always produces a result for 2 or more training observations. In our application, ML fails completely for fewer than 7 training observations, as it was used with 6 feature variables (i.e. the three pattern spectrum moments from each of the foreground and background images).

We use the ML and MLP classifiers and the minimum distance (MD) classifier [85] (which can be used regardless of sample size), as benchmark comparisons for the PEF. Unlike ML, MLP and PEF, which use the pattern spectrum moments as feature variables, the MD method is applied to the discrete pattern spectrum itself (and all pattern spectra are filled out with zeroes to the size of the largest pattern spectrum observed, to make this possible). MD chooses class k such that

$$k = \underset{t}{\text{argmin}} \ \{\Sigma_{r=0}^{R}[\Phi_{obs}(r) - \Phi_t(r)]^2\}, \tag{9.13}$$

where r indexes the scale of the SE used in opening and R is the largest such scale, and $\Phi_{obs}(r)$ and $\Phi_t(r)$ are the values of the pattern spectrum at scale r for the observed image and the t-th class respectively. In practice we have a sum of squared distances, as in Eq. (9.13), for both the foreground and background pattern spectra and we classify using the total of these sums. See [85] for optimal properties of the MD classifier.

9.6.4. Results for simulated images

The results below use an octagonal SE for the opening of the binary octagonal grain images, and a pyramidal SE calculated using the fast algorithm of [87] for the gray-scale pyramid images. The gradient of the slope of the pyramid SE was decreased from very steep steps of 31 to steps of 2 (almost flat), with height being kept at approximately 255 by increasing base size as the gradient decreases. In this case, in calculation of the moments in Eq. (9.4), r represents the gradient.

Performance is assesssed through several error measures, referred to as type 0, type 1, type 2 and type 3 error. Type i error is defined as

$$\frac{1}{n}\Sigma_{j=1}^{n} I_i(C_{est}^{j} - C_{obs}^{j}),\ \ i = 0, \ldots, 3, \tag{9.14}$$

where j indexes the n images to be classified, C_{est}^{j} and C_{obs}^{j} are respectively the estimated and actual class/state of image j, and $I_i(x) = 1$ if $\mid x \mid > i$, else $I_i(x) = 0$. Hence type i error records the proportion of images not classified to within i units of their actual time state. These error measures reflect the nature of the corrosion application. For applications where safety or damage limitation is critical, the maximum allowable chance of a component or system failing is set deliberately low to allow for uncertainty in safety status and ensure safe operation. It is more important always to predict the safety/damage status close to the correct class, than to be exactly correct most of the time but have the occasional large error. Where some overlap between classes is also possible, such as in the simulated images, for which the evolution parameters controlling the grain size and density are random variables, classification may not be exactly correct, but still effective for assessing evolution state. In assessing corrosion, deciding whether corrosion is too advanced will depend on being ‘close’ to the time classes representing advanced corrosion in the training data. Again, allowing some tolerance before recording an error is sensible.

Table 9.1 and Figure 9.12 show results for the binary octagons, for training sets of size 30 and 5 images at each of times $t = 0, \ldots, 20$. The smaller number of training images was chosen as in many real world quality control situations very limited training samples are available. The available data is often limited by difficulties in process monitoring, tracking of parts, record keeping, and various other factors. In both cases, an independent test set of 30 images was used at each time step to find the error rates, so that $n = 630$ in Eq. (9.14). The smaller sample size is too small for the unpooled ML method to be feasible. For the gray-scale pyramids, the errors have been averaged over 10 sets of training observations of a given size, for greater stability with smaller training set sizes. Table 9.1 and Figure 9.13 give the results.

No method achieves a low type 0 error, and classifying exactly correctly is clearly difficult. For both octagons and pyramids, PEF is better than MD

Training set size	30 images				5 images			
Error type	0	1	2	3	0	1	2	3
Images	Binary octagons							
Method	% error							
MLP	39	5.9	1.3	0.3	45.6	8.7	1.9	0.5
ML	38.7	3.5	0.6	0.2	-	-	-	-
MD	49	11	1.9	0.2	57	16	5.4	1.1
PEF	49	5.4	0.6	0	47.6	5.6	0.6	0
Images	Gray-scale pyramids							
Method	% error							
MLP	42.3	16.9	7.8	2.6	46.1	17.1	8.6	3.6
ML	50.1	5.3	0.4	0.0	-	-	-	-
MD	61.0	23.0	9.4	4.4	66.3	26.5	11.4	5.3
PEF	60.4	11.4	2.1	0.4	60.0	11.4	2.3	0.4

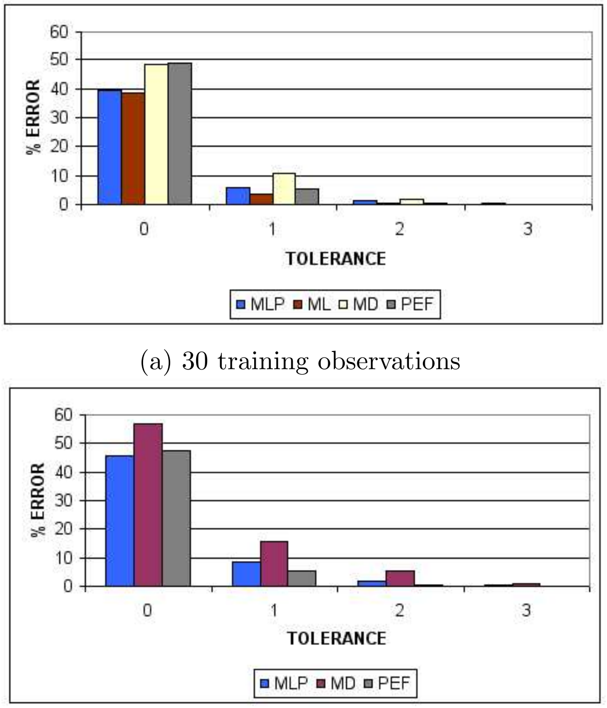
Table 9.1. Type 0, type 1, type 2 and type 3 errors for binary octagonal textures and gray-scale pyramid textures, using 30 and 5 training images.

for all error types, especially for the smaller training set, and better than MLP for all except type 0 error. PEF is barely affected by a reduction in training set size, whereas MD and MLP both perform better for the larger training set.

For the octagons, and 30 training observations, ML is better than PEF for type 0 and 1 errors, but ML is unavailable for the smaller training set sizes, and PEF and ML are very similar for error types 2 and 3. For the pyramids, ML is always better than PEF for the larger training sample, but again cannot be used for smaller samples.

Figures 9.14 and 9.15 show type 0 and type 2 errors as a function of training set size. For type 0 error, for the octagons PEF is best for very small numbers of observations, beyond which MLP is best, whereas PEF is poorer than the other methods for the pyramids. Where some tolerance is allowed, PEF is clearly the best and most consistent choice regardless of sample size.

Further results, for triangular grains and mixed grain types, are given in [54, 55]. Robustness of the PEF, when trained on one set of textures and applied to textures in which only the grain shape differed but the generative model followed the same evolution process, was also investigated in [54, 55]. Achieving acceptable results depends on the relationships of the moments with time being similar for the two processes.



(a) 30 training observations

(b) 5 training observations

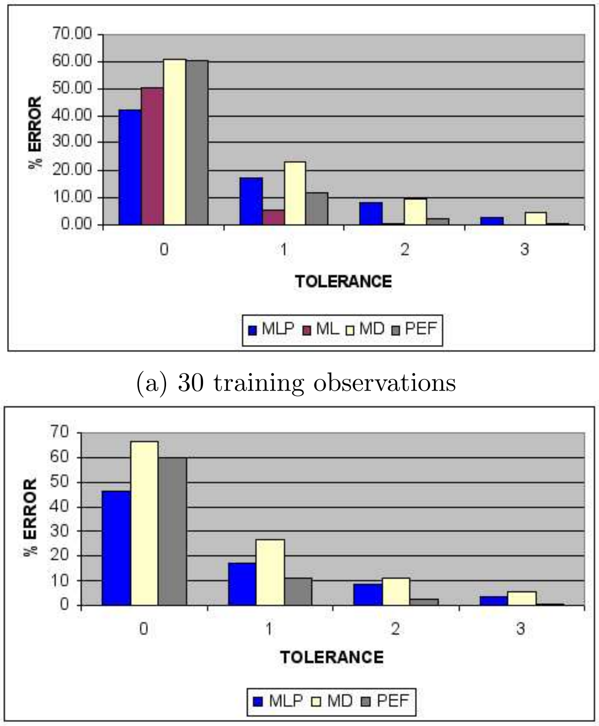
Figure 9.12. Percentage error versus error type (tolerance) for binary octagons.

9.6.5. Results for corrosion images

The granulometric opening of the corrosion images here uses gray-scale pyramid structuring elements of increasing base size as 3-D probing functions, applied as in Section 9.6.4.

For the simulated training images generated from a model, the evolution parameters used in the PEF approach are known. The model for the simulated images is used as an estimate of the model linking the parameters to time for the corrosion images. It is not necessary that the parameters relate to precise physical structures in the real images. They are instead "virtual" parameters which are used to encode time.

The ten 1400×1400 corrosion images (Figure 9.8) were each cut into 49 sub-images of size 200×200. The first 7 such images (from the top row) for each time were used as the training data, reflecting the small training set sizes typically available in practice for such applications, and another 7 images from the right hand side of the plate were considered as independent test data. The limited number, and the position, of the selected images owes to the reflection of the camera on the polished surface (seen in Figure 9.8) being harder to remove than the gradual light variations. However, there



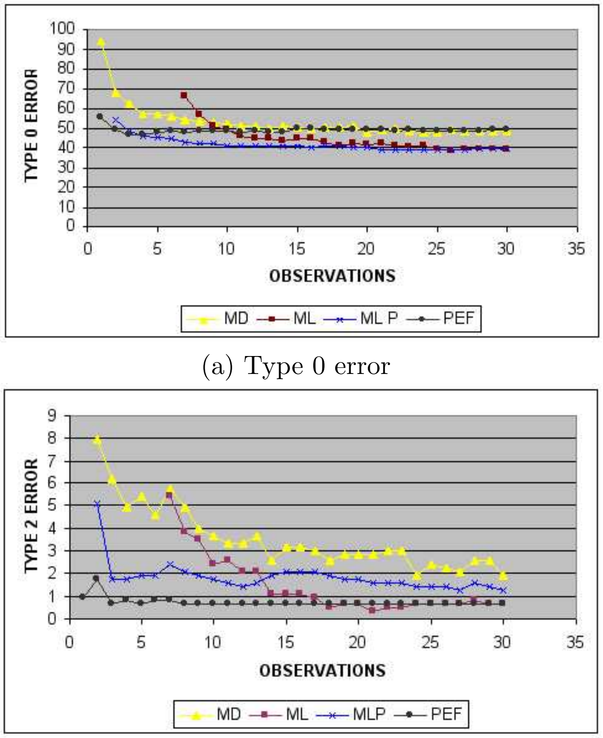
(a) 30 training observations

(b) 5 training observations

Figure 9.13. Percentage error versus error type (tolerance) for gray-scale pyramids.

is substantial variation within the corrosion images at a given time t, and obtaining sufficiently similar training and test images was difficult. Figure 9.16 ilustrates this variation, showing a 200 × 200 training image taken from the top left of the original image at day 5, and a test image taken from the bottom right. In this application, clearly larger training images are desirable, to encompass the variation found in these stochastic textures at a given point in time. The results shown are therefore for the dependent data.

Figures 9.17 and 9.18 show percentage errors for 1 to 7 training images. Given the small training set sizes, the ML method is not used and MLP is inapplicable for only 1 training image. The error rates shown are averages. The training observations were chosen randomly from the available set of training images, and up to 10 such choices were made, depending on the training set size. Averaging the resulting error rates smooths out fluctuations caused by the particular choice of small training set. Results are shown for the combined PEF method, as previously, as well as for PEF applied to the image foreground (FRONT) and background (BACK) separately. The results are compared with those of MD and MLP.



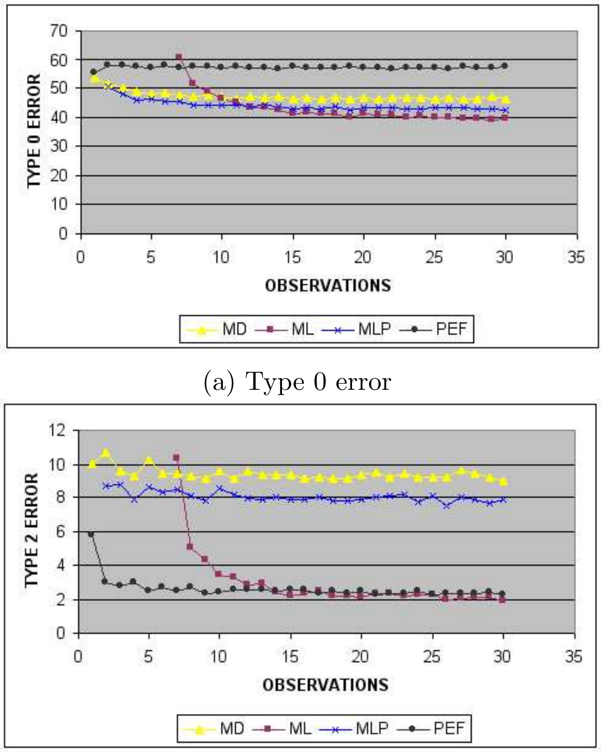
(a) Type 0 error

(b) Type 2 error

Figure 9.14. Percentage error versus training set size for binary octagons.

For type 0 and type 1 errors, MD is best for 1 or 2 observations. For type 0 error, MLP is clearly best for more than 2 observations and PEF, FRONT and BACK methods are poorer. For type 1 errors, MLP and PEF are equally good choices for more than 2 observations. For type 2 and type 3 errors, MD is best only for 1 observation, and beyond that PEF and FRONT are generally best. BACK does not perform as well, and of the combined PEF, FRONT, and BACK, the combined PEF is the most consistent of the three methods. These results are not quite as good as for the simulated images, however PEF is still the best method overall when a small tolerance is allowed, where there is more than 1 training observation. MD and MLP are more accurate with no tolerance.

PEF is more sensitive to changes in texture than MLP and MD, which is generally an advantage, although it does require that the moments and evolution functions are similar for the training data and the process generating the textures to be classified. The poorer performance of PEF for the real data with only 1 or 2 training observations is a result of the larger



(a) Type 0 error

(b) Type 2 error

Figure 9.15. Percentage error versus training set size for gray-scale pyramids.

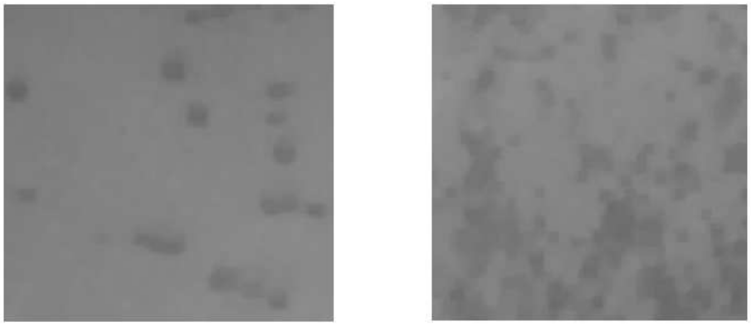
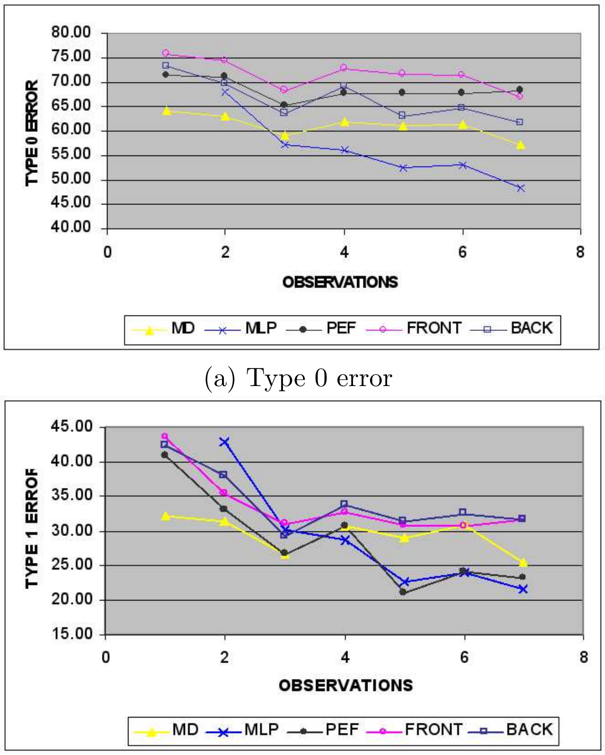


Figure 9.16. Training image (left) and independent image (right) from day 5.

variation in the moments within a given time state requiring more data to fit the regression model reliably.



(a) Type 0 error

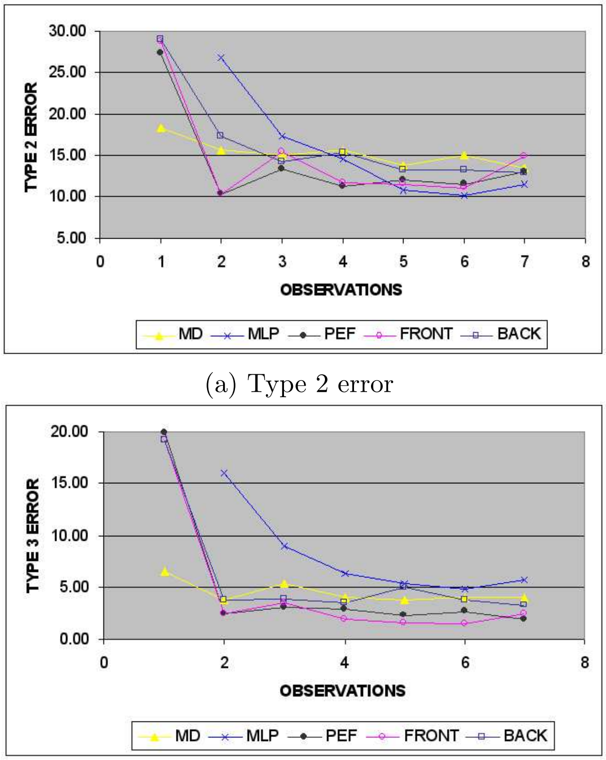
(b) Type 1 error

Figure 9.17. Percentage error versus number of training observations for corrosion images.

9.7. Summary

Morphological methods have the advantage, over other approaches to texture analysis, of directly evaluating shape and size information in the texture image. Granulometric analysis applies morphological operators in sequence at a series of different scales. The resulting granulometric pattern spectrum and its statistical summaries characterize the proportions of the image containing objects/texture elements of different sizes and shapes. Evolving shape-based textures, arising from a process in which size and/or shape of texture elements present changes with time, are therefore intrinsically well suited to a granulometric approach to texture analysis.

Furthermore, most texture classification techniques allocate images to one of a discrete set of unrelated static textures, rather than placing an image directly to a point on an underlying continuous scale on the basis of a set of evolving textures related by time.



(a) Type 2 error

(b) Type 3 error

Figure 9.18. Percentage error versus number of training observations for corrosion images.

The PEF method uses both granulometry and the dynamic nature of the textures, to relate the values of the hidden parameters of an underlying generative image model to the observable pattern spectrum moments, using a statistical model. It then uses this model to estimate directly the evolution status of a new image.

The results shown clearly indicate that using the relationship between the moments and evolution parameters, rather than granulometric information alone, improves classification accuracy for small to moderate training samples, and gives consistent performance for larger training samples.

The PEF method has been shown to perform better than other classifiers for small training sets when the observed data is visually similar to the training data or is controlled by a similar random growth process. This is especially the case when the main priority is for the predicted class to be close to the correct class, if not exactly correct all of the time. PEF performs well when only a few training observations are available, due to its use of information from all time states, and, unlike the maximum likelihood (ML) method, it does not require a minimum number of observations. Compared to both the minimum distance (MD) [85] and pooled maximum likelihood (MLP) [30] classifiers, it performs well, hardly ever deviating far from the correct class. This is a great advantage in quality control applications.

As it predicts directly to a time state on a continuous scale, PEF also has the advantage of being able to predict intermediate time states between those represented in the training set. Since it involves multiple linear regression, reliable prediction of time state does require that the training data covers the period of time or severity of interest, to avoid extrapolation [99].

While granulometries are non-linear, the PEF modeling linearly relates the granulometric moments to the evolution parameters. This facilitates inversion in order to estimate the evolution parameters for a new image. It is assumed that any departures from this model or from the required conditions for a multiple linear regression will not seriously impact on the accuracy of prediction. However, more generally, non-linear regression models [73] or time series methods [15] could be used for modeling. Time series models are likely to make back-prediction, as used in the PEF approach, difficult, but are worth considering as the best form of model may depend on the application.

Possible improvements to PEF include using a different point process, i.e. more varied and irregular shapes of grain to build the evolution model, and/or different grain density distributions, and different growth functions for the mean grain size and its variance. Regarding the point process, it is likely that new sites of corrosion will depend on the existing corrosion rather than being randomly distributed. A suitable model should result in model parameters being more closely related to the granulometric moments of the real data and therefore more accurate prediction of time state. Optimizing choice of SE(s) [23], or use of multiple SEs in a more complex granulometry [5], are also worth considering.

9.8. Acknowledgments

We gratefully ackowledge the inspiration, encouragement and advice of Professor Edward R. Dougherty of the Dept. of Electrical Engineering, Texas A&M University. We are also grateful to the Engineering and Physical Sciences Research Council and Rolls Royce plc, Derby, England, for the CASE studentship which supported the Ph.D. studies of Jennifer McKenzie.

Bibliography

[1] G. Ayala and J. Domingo, Spatial size distributions: applications to shape and texture analysis, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23(12), pp. 1430–1442, 2001.

[2] A. Bangham and S. Marshall, Image and signal processing with mathematical morphology, *IEE Journal of Electronics and Communication*, vol. 10(3), pp. 117–128, 1998.

[3] S. Baeg, S. Batman, E.R. Dougherty, V.G. Kamat, N. Kehtarnavaz, K. Seunghan, A. Popov, K. Sivakumar and R. Shah, Unsupervised morphological granulometric texture segmentation of digital mammograms, *Journal of Electronic Imaging*, vol. 8(1), pp. 65–75, 1999.

[4] R. Bajcsy, Computer description of textured surfaces, *Proceedings of 3rd International Joint Conference on Artificial Intelligence*, pp. 572–579, 1973.

[5] S. Batman and E.R. Dougherty, Size distributions for multivariate morphological granulometries: texture classification and statistical properties, *Optical Engineering*, vol. 36(5), pp. 1518–1529, 1997.

[6] P. Brodatz, *Textures*, Dover, New York, 1966.
Images available at http://www.ux.his.no/∼tranden/brodatz/

[7] J.R. Carr and F. Pellon de Miranda, The semivariogram in comparison to the co-occurrence matrix for classification of image texture, *IEEE Transactions on Geoscience and Remote Sensing*, vol. 36(6), pp. 1945–1952, 1998.

[8] M.J. Chantler and J.P. Stoner, Automatic interpretation of sonar image sequences using temporal feature measures, *IEEE Journal of Oceanic Engineering*, vol. 22(1), pp. 47–56, 1997.

[9] R. Chellappa and R.L. Kashyap, Texture synthesis using 2-D non-causal autoregressive models, *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 33, pp. 194–203, 1985.

[10] Y. Chen, E.R. Dougherty, S.M. Totterman and J.P. Hornak, Classification of trabecular structure in magnetic resonance images, *Magnetic Resonance Medicine*, vol. 29(3), pp. 358–370, 1993.

[11] Y. Chen and E.R. Dougherty, Gray-scale morphological granulometric texture classification, *Optical Engineering*, vol. 33(8), pp. 2713–2722, 1994.

[12] Y. Chen and E.R. Dougherty, Logical structure filters, *Optical Engineering*, vol. 37(6), pp. 1668–1676, 1998.

[13] S. Chindaro, K. Sirlantzis and M.C. Fairhurst, Component based feature space partition and combination in multiple colour spaces for texture classification, *Proceedings of VIE2005, IEE International Conference on Visual Information Engineering, Glasgow*, Institution of Electrical Engineers, London, pp. 211–218, 2005.

[14] D.A. Clausi, Comparison and fusion of co-occurrence, Gabor and MRF texture features for classification of SAR sea-ice imagery, *Atmosphere-Ocean*, vol. 39(3), pp. 183–194, 2001.

[15] W.S. Cleveland, *Visualizing Data*, Hobart Press, Summit, New Jersey, USA, 1993.

[16] R.W. Conners and C.A. Harlow, A theoretical comparison of texture algorithms, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 2(3), pp. 204–222, 1980.

[17] T.J. Dennis and N.G. Dessipris, Fractal modeling in image texture analysis, *IEE Proceedings F*, vol. 136(5), pp. 227–235, 1989.

[18] P.J. Diggle, Binary mosaics and the spatial pattern of heather, *Biometrics*, vol. 37, pp. 531–539, 1981.

[19] E.R. Dougherty, *Random Processes for Image and Signal Processing*, SPIE/IEEE Series on Imaging Science and Engineering, 1998.

[20] E.R. Dougherty and J. Astola, *An Introduction to Nonlinear Image Processing*, vol. TT 16, SPIE Press, Washington, USA, 1994.

[21] E.R. Dougherty and J. Astola, *Non-linear filters for image processing*, SPIE/IEEE series on Image Science and Engineering, SPIE Press, Washington, USA, 1999.

[22] E.R. Dougherty and Y. Chen, Logical structure filters, *Optical Engineering*, vol. 37(6), pp. 1668–1676, 1998.

[23] E.R. Dougherty and R.P. Loce, Efficient design strategies for the optimal binary digital morphological filter: probabilities, constraints, and structuring-element libraries, *Mathematical Morphology in Image Processing*, Ch. 2, Marcel Dekker Inc., New York, 1992.

[24] E.R. Dougherty and R.A. Lotufo, *Hands-on Morphological Image Processing*, SPIE Press, Washington, USA, 2003.

[25] E.R. Dougherty and J.B. Pelz, Morphological granulometric analysis of electrophotographic images – size distribution statistics for process control, *Optical Engineering*, vol. 30(4), pp. 438–445, 1991.

[26] E.R. Dougherty, J.T. Newell and J.B. Pelz, Morphological texture-based maximum-likelihood pixel classification based on local granulometric moments, *Pattern Recognition*, vol. 25(10), pp. 1181–1198, 1992.

[27] E.R. Dougherty, J.B. Pelz, F. Sand and A. Lent, Morphological image segmentation by local granulometric size distributions, *Journal of Electronic Imaging*, vol. 1(1), pp. 46–60, 1992.

[28] A. Drimbarean and P.F. Whelan, Experiments in color texture analysis, *Pattern Recognition Letters*, vol. 22, pp. 1161–1167, 2001.

[29] R.O. Duda and P.E. Hart, *Pattern Recognition and Scene Analysis*, John Wiley and Sons, New York, 1973.

[30] B.S. Everitt and G. Dunn, *Applied Multivariate Data Analysis*, Edward Arnold, London, 1991.

[31] S. Fogel and D. Sagi, Gabor filters as texture discriminators, *Biological Cybernetics*, vol. 61, pp. 103–113, 1989.

[32] M.M. Galloway, Texture analysis using gray level run lengths, *Computer Graphics and Image Processing*, vol. 4, pp. 172–179, 1975.

[33] P. Garcia-Sevilla and M. Petrou, Analysis of irregularly shaped texture regions, *Computer Vision and Image Understanding*, vol.84, pp. 62–76, 2001.

[34] R. Gnanadesikan, *Methods for Statistical Data Analysis of Multivariate Observations*, John Wiley and Sons, pp. 90–96, 1977.

[35] J.F. Haddon and J.F. Boyce, Co-occurrence matrices for image analysis, *IEE Electronics and Communications Engineering Journal*, vol. 5(2), pp. 71–83, 1993.

[36] J.C. Handley and E.R. Dougherty, Maximum likelihood estimation for the two-dimensional discrete Boolean random set and function models using multidimensional linear samples, *Graphical Models and Image Processing*, vol. 59(4), pp. 221–231, 1997.

[37] R.M. Haralick and L.G. Shapiro, *Computer and Robot Vision 1*, Addison-Wesley, Reading, MA, 1992.

[38] R.M. Haralick, K. Shanmugan and I.H. Dinstein, Textural features for image classification, *IEEE Transactions on Systems, Man and Cybernetics*, vol. 3, pp. 610–621, 1973.

[39] G.M. Haley and B.S. Manjunath, Rotation invariant texture classification using a complete space-frequency model, *IEEE Transactions on Image Processing*, vol. 8, pp. 256–269, 1999.

[40] E. Hernández and G. Weiss, *A First Course on Wavelets*, CRC Press, New York, 1996.

[41] G.W. Horgan, Mathematical morphology for analysing soil structure from images, *European Journal of Soil Science*, vol. 49, pp. 161–173, 1998.

[42] G.W. Horgan, C.A. Reid and C.A. Glasbey, Biological image processing and enhancement, *Image Processing and Analysis, A Practical Approach*, R. Baldock and J. Graham, eds., Oxford University Press, Oxford, UK, pp. 37–67, 2000.

[43] B.B. Hubbard, *The World According to Wavelets: The Story of a Mathematical Technique in the Making*, A.K. Peters Ltd., Wellesley, MA, 1995.

[44] H. Iversen and T. Lonnestad. An evaluation of stochastic models for analysis and synthesis of gray-scale texture, *Pattern Recognition Letters*, vol. 15, pp. 575–585, 1994.

[45] A.K. Jain and F. Farrokhnia, Unsupervised texture segmentation using Gabor filters, *Pattern Recognition*, vol. 24(12), pp. 1167–1186, 1991.

[46] T. Jossang and F. Feder, The fractal characterization of rough surfaces, *Physica Scripta*, vol. T44, pp. 9–14, 1992.

[47] A.K. Katsaggelos and T. Chun-Jen, Iterative image restoration, *Handbook of Image and Video Processing*, A. Bovik, ed., Academic Press, London, pp. 208–209, 2000.

[48] M. Köppen, C.H. Nowack and G. Rösel, Pareto-morphology for color image processing, *Proceedings of SCIA99, 11th Scandinavian Conference on Image Analysis 1, Kangerlussuaq, Greenland*, pp. 195–202, 1999.

[49] S. Krishnamachari and R. Chellappa, Multiresolution Gauss-Markov random field models for texture segmentation, *IEEE Transactions on Image Processing*, vol. 6(2), pp. 251–267, 1997.

[50] T. Kurita and N. Otsu, Texture classification by higher order local autocorrelation features, *Proceedings of ACCV93, Asian Conference on Computer Vision, Osaka*, pp. 175–178, 1993.

[51] S.T. Kyvelidis, L. Lykouropoulos and N. Kouloumbi, Digital system for detecting, classifying, and fast retrieving corrosion generated defects, *Journal of Coatings Technology*, vol. 73(915), pp. 67–73, 2001.

[52] Y. Liu, T. Zhao and J. Zhang, Learning multispectral texture features for cervical cancer detection, *Proceedings of 2002 IEEE International Symposium on Biomedical Imaging: Macro to Nano*, pp. 169–172, 2002.

[53] G. McGunnigle and M.J. Chantler, Modeling deposition of surface texture, *Electronics Letters*, vol. 37(12), pp. 749–750, 2001.

[54] J. McKenzie, S. Marshall, A.J. Gray and E.R. Dougherty, Morphological texture analysis using the texture evolution function, *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 17(2), pp. 167–185, 2003.

[55] J. McKenzie, *Classification of dynamically evolving textures using evolution functions*, Ph.D. Thesis, University of Strathclyde, UK, 2004.

[56] S.G. Mallat, Multiresolution approximations and wavelet orthonormal bases of $L^2(R)$, *Transactions of the American Mathematical Society*, vol. 315, pp. 69–87, 1989.

[57] S.G. Mallat, A theory for multiresolution signal decomposition: the wavelet representation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 674–693, 1989.

[58] B.S. Manjunath and W.Y. Ma, Texture features for browsing and retrieval of image data, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, pp. 837–842, 1996.

[59] B.S. Manjunath, G.M. Haley and W.Y. Ma, Multiband techniques for texture classification and segmentation, *Handbook of Image and Video Processing*, A. Bovik, ed., Academic Press, London, pp. 367–381, 2000.

[60] G. Matheron, *Random Sets and Integral Geometry*, Wiley Series in Probability and Mathematical Statistics, John Wiley and Sons, New York, 1975.

[61] J. Miles and M. Shevlin, *Applying Regression and Correlation: A Guide for Students and Researchers*, SAGE Publications Ltd, London, 2001.

[62] M. Mirmehdi and M. Petrou, Segmentation of color textures, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22(2), pp. 142–158, 2000.

[63] A. Monadjemi, B.T. Thomas and M. Mirmehdi, Speed v. accuracy for high resolution colour texture classification, *Proceedings of 13th British Machine Vision Conference*, BMVA Press, pp. 143–152, 2002.

[64] P. Moulin, Multiscale image decompositions and wavelets, *Handbook of Image and Video Processing*, A. Bovik, ed., Academic Press, London, pp. 289–300, 2000.

[65] W.M. Mullins, E.J. Shumaker and G.J. Tyler, Stochastic kinetics of corrosion and fractal surface evolution, *The Journal of Corrosion Science and Engineering*, vol. 1(7), 1997. (At http://www.jsce.org/Volume1/paper7/v1p7.html)

[66] T. Ojala, M. Pietikinen and D. Harwood, A comparative study of texture measures with classification based on feature distributions, *Pattern Recognition*, vol. 29, pp. 51–59, 1996.

[67] M. Petrou and G. Lazaridis, Texture segmentation using local Walsh coefficients, *Proceedings of VIE2005, IEE International Conference on Visual Information Engineering, Glasgow*, Institution of Electrical Engineers, London, pp. 189–194, 2005.

[68] O. Pichler, A. Teuner and B.J. Hosticka, A comparison of texture feature extraction using adaptive Gabor filter, pyramidal and tree structured wavelet transforms, *Pattern Recognition*, vol. 29(5), pp. 733–742, 1996.

[69] R. Porter and N. Canagarajah, A robust automatic clustering scheme for image segmentation using wavelets, *IEEE Transactions on Image Processing*, vol. 5(4), pp. 662–665, 1996.

[70] K. Rajesh, C.V. Jawahar, S. Sengupta, S. and S. Sinha, Performance analysis of textural features for characterization and classification of SAR images, *International Journal of Remote Sensing*, vol. 22(8), pp. 1555–1569, 2001.

[71] T. Randen and J.H. Husoy, Multichannel filtering for image texture segmentation, *Optical Engineering*, vol. 33(8), pp. 2617–2625, 1994.

[72] T. Randen and J.H. Husoy, Filtering for texture classification: a comparative study, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21(4), pp. 291–310, 1999.

[73] J.O. Rawlings, *Applied Regression Analysis: A Research Tool*, Wadsworth and Brooks/Cole, Statistics/Probability Series, Belmont, CA, 1988.

[74] S. Rebbapragada, M.J. Palakal, and R.M. Pidaparti, Corrosion detection and quantification using image processing for aging aircraft panels, *Proceedings of the 3rd Joint FAA/DoD/NASA Conference on Ageing Aircraft, Albuquerque, NM*, 1999.

[75] T.R. Reed and J.M.H. du Buf, A review of recent texture segmentation and feature extraction techniques, *CVGIP: Image Understanding*, vol. 57(3), pp. 359–372, 1993.

[76] A. Rosenfeld, Multi-resolution image representation, *Digital Image Analysis*, E.S. Levadi, ed., Pitman Press, pp. 22–25, 1984.

[77] F. Sand and E.R. Dougherty, Asymptotic normality of the morphological pattern-spectrum moments and orthogonal granulometric generators, *Journal of Visual Communication and Image Representation*, vol. 3(2), pp. 203–214, 1992.

[78] F. Sand and E.R. Dougherty, Asymptotic granulometric mixing theorem: morphological estimation of sizing parameters and mixture proportions, *Pattern Recognition*, vol. 31(1), pp. 53–61, 1998.

[79] J. Serra, *Image Analysis and Mathematical Morphology*, Academic Press, New York, 1982.

[80] J. Serra, *Image Analysis and Mathematical Morphology, Volume 2: Theoretical Advances*, Academic Press, London, 1988.

[81] J. Serra and G. Verchery, Mathematical morphology applied to fibre composite materials, *Film Science and Technology*, vol. 6, pp. 141–158, 1973.

[82] L. Shafarenko, M. Petrou and J. Kittler, Automatic watershed segmentation of randomly textured color images, *IEEE Transactions on Image Processing*, vol. 6(11), pp. 1530–1543, 1997.

[83] M. Singh and S. Singh. Spatial texture analysis: a comparative study. At http://citeseer.ist.psu.edu/507622.html

[84] S. Singh and M. Sharma, Texture analysis experiments with Meastex and Vistex Benchmarks, *Proceedings of International Conference on Advances in Pattern Recognition*, Lecture Notes in Computer Science vol. 2013, Springer, pp. 417–424, 2001 (http://citeseer.ist.psu.edu/528275.html).

[85] K. Sivakumar and J. Goutsias, Morphologically constrained GRFs: applications to texture synthesis and analysis, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21(2), pp. 99–113, 1999.

[86] K. Sivakumar, Y. Balagurunathan and E.R. Dougherty, Asymptotic joint normality of the granulometric moments, *Pattern Recognition Letters*, vol. 22, pp. 1537–1543, 2001.

[87] K. Sivakumar, M.J. Patel, N. Kehtarnavaz, Y. Balagurunathan and E.R. Dougherty, A constant-time algorithm for erosions/dilations with applications to a morphological texture feature computation, *Real Time Imaging*, vol. 6, pp. 223–239, 1999.

[88] L.-K. Soh and C. Tsatsoulis, Texture analysis of SAR sea ice imagery using gray level co-occurrence matrices, *IEEE Transactions on Geoscience and Remote Sensing*, vol. 37(2), pp. 780–795, 1999.

[89] K.Y. Song, J. Kittler and M. Petrou, Defect detection in random colour textures, *Image and Vision Computing*, vol. 14(9), pp. 667–683, 1996.

[90] J. Teuber, *Digital Image Processing*, Series in Acoustics, Speech and Signal Processing, Prentice Hall, New York, 1993.

[91] N. Theera-Umpon and P.D. Gader, Counting white blood cells using morphological granulometries, *Journal of Electronic Imaging*, vol. 9(2), pp. 170–177, 2000.

[92] F. Tomita and S. Tsuji, *Computer Analysis of Visual Textures*, Kluwer Academic Publishers, Dordrecht, Netherlands, 1990.

[93] K.R. Tretheway and P.R. Roberge, The characterization of surface profiles created by localized corrosion with stochastic and fractal analysis techniques, *Localized Damage*, vol. 3, pp. 323–330, 1994.

[94] M. Unser, Texture classification and segmentation using wavelet frames, *IEEE Transactions on Image Processing*, vol. 4(11), pp. 1549–1560, 1995.

[95] A.L. Vickers and J.W. Modestino, A maximum likelihood approach to texture classification, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 4, pp. 61–68, 1982.

[96] VisTex Database, MIT Media Library.
At http://vismod.media.mit.edu/vismod/imagery/VisionTexture/vistex.html

[97] J.S. Walker, *Fourier Analysis*, Oxford University Press, New York, 1988.

[98] C. Weiping and C. Xu, Fractal structure of uniform corrosion surface for Fe-based alloys, *Journal of Materials Science Letters*, vol. 16, pp. 113–114, 1997.

[99] G.B. Wetherill, *Regression Analysis with Applications*, Monographs on Statistics and Applied Probability, Chapman and Hall, London, 1986.

[100] D.S. Wickramanayake, E.A. Edirisinghe and H.E. Bez, A wavelet based image quilting approach to fast, multiresolution texture synthesis, *Proceedings of VIE 2005, IEE International Conference on Visual Information Engineering, Glasgow*, Institution of Electrical Engineers, London, pp. 93–100, 2005.

[101] T. Xiaoou, Texture information in run-length matrices, *IEEE Transactions on Image Processing*, vol. 7(11), pp. 1603–1609, 1998.

[102] L. Yih-Gong, L. Jia-Hong and H. Yuang-Cheh, Genetic-based fuzzy hit-or-miss texture spectrum for texture analysis, *Electronics Letters*, vol. 31(23), pp. 1986–1987, 1995.

[103] J. Zhang, P. Fieguth and D. Wang, Random Field Models, *Handbook of Image and Video Processing*, (A. Bovik, ed.), Academic Press, London, 2000.

[1] Department of Statistics and Modeling Science, University of Strathclyde, Livingstone Tower, 26 Richmond Street, Glasgow, G1 1XH, U.K.; alison@stams.strath.ac.uk

[2] Department of Electrical and Electronic Engineering, University of Strathclyde, Royal College, 204 George Street, Glasgow, G1 1XW, U.K.; s.marshall@eee.strath.ac.uk