Segmentation of random textures by morphological and linear operators

Aurélien Cord,[a] Dominique Jeulin,[a] and Francis Bach[a]

[a] *Centre de Morphologie Mathmatique (CMM), 35 rue Saint Honoré, 77305 Fontainebleau CEDEX, France*
aurelien.cord@cmm.ensmp.fr

Abstract We propose a linear and a morphological approach for the characterization and segmentation of binary and digital random textures. We focus on descriptors at the level of pixels in images, combined with statistical learning to select and weight them. The approach is illustrated on simulations of textures patchworks, for which errors of classification can be evaluated.

Keywords: Texture Segmentation, Random Textures, Morphological Operations, Multivariate Statistical Analysis, Linear Discriminant Analysis, Machine Learning.

1. Introduction

Image segmentation is a common and important task aimed at extracting objects that can be subsequently measured or analyzed. This is in particular useful when looking for defects in materials or for pathological cells in a biological tissue. In many cases, the background and the objects themselves are non uniform and made of so-called "textures". A challenging problem concerns the automatic extraction (or segmentation) of various textures present in the same image.

For this purpose, a variety of statistical techniques have been used, such as histogram based texture analysis techniques corresponding to the use of co-occurrence matrices [7], texture modeling [4], filtering approaches [11] and wavelet transformations of images [10, 12, 15]. However on materials displaying complex patterns that are random in appearance (i.e. not periodic), segmenting texture turns out to be difficult [18].

To handle it, we propose the following approach: to describe the texture for each pixel of the image, accounting for morphological information in its neighborhood at different scales and then using a multivariate statistical approach. This approach is illustrated and validated for binary and gray level textures through simulations.

2. Theoretical approach of random textures

Natural textures have the following common characteristics: a texture usually shows fluctuations at a small scale, and some uniformity at a large scale. The presence of fluctuations requires the use of a probabilistic approach to characterize textures. In this framework we will consider binary textures and gray level textures as realizations of random sets or of random functions. Therefore, from a theoretical point of view, a random texture is completely known from its Choquet capacity.

It can be shown [14] that a random closed set A is known from the Choquet capacity functional $T(K)$ defined on compact sets K:

$$T(K) = 1 - P\{K \subset A^C\} \tag{1.1}$$

where P is the probability of the event $\{\}$. Similarly, an upper semi-continuous random function (RF) $Z(x)$ is characterized by the functional $T(g)$, defined for test functions g with a compact support K [8, 9]:

$$T(g) = P\{x \in D_Z(g)\} \tag{1.2}$$

with

$$D_Z(g)^c = \{x \ , \ Z(x+y) < g(y) \ , \ \forall \ y \in K\} \tag{1.3}$$

When the compact set K is a point x and $g(x) = z$, the cumulative distribution function is obtained. When using the two points $\{x, x+h\}$ and the two functions $g(x) = z_1$ and $g(x+h) = z_2$ we can derive the bivariate distribution $F(z_1, z_2, h)$. More generally, using $g(x) = z$ for $x \in K$ and $g(x) = +\infty$ for $x \notin K$ we obtain the distribution function of $Z(x)$ after a change of support according to the sup over any compact set K.

From this theoretical background, it turns out that good candidates for texture descriptors can be provided by estimates of probabilities obtained after dilations (or erosions) by compact sets for binary textures. Similarly, distribution functions after dilations (or erosions) of gray level images will provide texture descriptions in the digital case. Changing the shape and size of the compact set K provides us with a full set of data.

From a practical point of view, estimates of the Choquet capacity $T(K)$ or $T(g)$ can be obtained on images. The pertinence of descriptors will depend on the statistical precision of estimates. In a second step, a selection of the more efficient descriptors must be made. This is usually performed by multivariate analysis. This approach was successfully applied to the classification of images in the standard case when there is a single texture per field of view [1, 6]. To address the problem of segmentation of textures in images, a classification of pixels must be performed. This requires a local characterization, that can be made in different ways:

1. Transform the image by dilations, erosions by K_i, g_i ("pixel" approach), and generation of a multispectral image from the collection of K_i or g_i.

2. Consider a neighborhood $B(x)$ of each pixel, and use a local estimate of $T(K)$, $T(g)$ inside $B(x)$. From the estimates, generate a multispectral image from the collection of K_i or g_i.

3. Use measures μ_i with a compact support K_i and estimate $\mu_i(A)$ or $\mu_i(Z)$, generating a multispectral image from the collection of K_i.

The first way is deterministic, and provides a set of transformed images (binary, or gray level). Combining erosions and dilations provides us filters, like granulometries, as already proposed [1, 6, 17]. This point is developed below in the section 3. The second way uses a local estimate of the Choquet capacity. It requires the appropriate choice of B, and will be illustrated in the application below (section 3.3). As particular cases of the third way are recovered various types of linear filters, like multi-scale convolution by Gaussian kernels, wavelets, curvelets, but also local measurements of the Minkowski functionals for a random set A (section 3.1).

3. Pixel texture description

In this work, texture properties are calculated for each pixel taking into account local properties of its neighborhood at varying scales. This provides us with 3 dimensional data having two spatial dimensions and a descriptor dimension. This allows us to characterize texture at the pixel level and follows the texton approach of [13]. Two families of descriptors are used: curvelets and morphological transformations.

3.1 The curvelet transform

The curvelet transform is a higher dimensional generalization of the wavelet transform, designed to represent images at different scales and different angles [3]. The use of this tool to characterize the texture in an image is recent [5]. Curvelets have very interesting properties in the context of object detection, in particular curved singularities can be well approximated with very few coefficients. This makes the curvelets coefficients for pixels belonging to a particular object very specific.

The curvelet filter bank is in essence a set of bandpass filters with range and orientation selective properties. Typically, we apply a linear filtering of each 100×100 neighborhood of every pixel by curvelets with different frequencies and orientations. The filter bank is decomposed into 4 sets of frequency containing 1, 8, 16 and 1 filters of varied orientations from the smallest frequency to the largest one. A spatial filtering of each of the 26 filters results is applied to calculate the local energy function. It aims to

identify areas where the band pass frequency components are strong, after conversion into gray levels. The outputs of this function are a first class of texture descriptors.

3.2 Morphological transformations

The morphological operators are connected to the description of random sets and of random functions by the Choquet capacity, as recalled in section 2. They consist of non linear image transformations [16]. Applying a succession of erosions (respectively dilation) by structuring elements K of increasing sizes makes progressively disappear characteristics of the images [2]. It is on this morphological property that the descriptor is based.

Different structuring elements are chosen: disks, vertical and horizontal segments. For a given type of structuring element and a list of sizes, all eroded and dilated images are evaluated. The descriptor is obtained by calculating the difference between the eroded images at the step n and $n+1$ as well as between the dilated images at steps $n+1$ and n. Therefore each pixel is described by a vector with k morphological components, where $k =$ number of sizes $\times$ number of structuring elements $\times$ 2 operations.

More complex pixel descriptors can also be proposed by using opening and closing operations instead of erosions and dilations, respectively, as made in [1,6,17]. The opening, obtained by an erosion followed by a dilation by a given structuring element, preserves bright parts of the image which can contain the structuring element, while the closing operation (dilation followed by an erosion) preserves dark parts.

In this study, we choose the following structuring elements:

- Small scales, erosion and dilation with series of sizes of $[1, 2, 3, 4, 5, 7, 9, 12] \times 2 + 1$ (size 48).
- Small scales, opening and closing with series of sizes of $[1, 2, 3, 4, 5, 7, 9, 12] \times 2 + 1$ (size 48),
- Large scales, opening and closing with series of sizes of $[2, 4, 8, 16, 32] \times 2 + 1$ (size 30),

3.3 Averaged descriptors

From the initial pixel descriptors, it is easy to estimate average descriptors in a window B, in order to use the local Choquet capacity. The optimal size of the window is obtained by minimizing the errors of classification of pixels. In what follows, averages in windows up to 60×60 are used.

4. Application

The present application is based on simulations of models of random textures. They are produced and mixed together. Using the knowledge of the

ground truth to calculate errors of classification, we are able to test different pixel descriptors for the segmentation. We produce a critical analysis of the approach and evaluate its limitations. Images used here are available on www.cmm.ensmp.fr/~cord/Synthetik_Texture/ for comparison with any other approach of segmentation.

4.1 Data

On the basis of [8,9], we simulate different textures in images of size 800×800 pixel (Fig. 1), using the Micromorph© software.

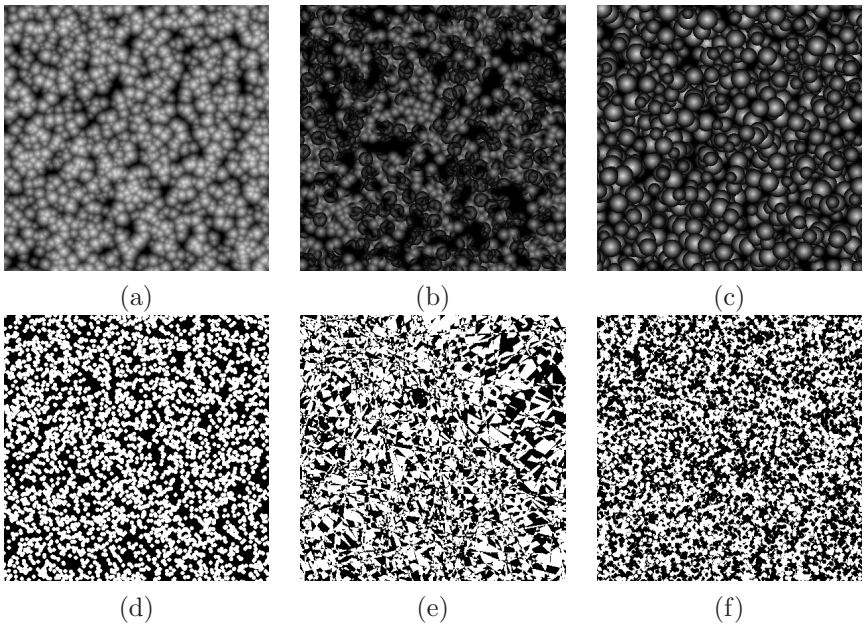


FIGURE 1. Simulated texture images. (a) Boolean random function with cone primary function. (b) Sequential Alternate random function with cone primary function. (c) Dead leaves random function. (d) Boolean random set, with disk primary grain (radius 6). (e) Binary dead leaves. (f) Poisson mosaic (from a Poisson line tesselation involving 400 lines).

- A boolean random function with cone primary function (Fig. 1.a), using a uniform distribution of radii between 1 and 32 pixels.
- A sequential alternate random function with cone primary function (Fig. 1.b)
- A dead leaves random function (Fig. 1.c)
- A boolean random set, with disk primary grain (radius 6) (Fig. 1.d)

- A binary dead leaves with disks (radius 6) (Fig. 1.e)
- A Poisson mosaic (from a Poisson line tessellation involving 400 lines (Fig. 1.f)
- A binary Poisson mosaic with 50 lines (Fig. 2.e), used as a mask.

We consider 4 case studies, corresponding to 4 patchwork images with a mixture of two textures. They are produced using two original images from Figures 1.a-f, combined using masks corresponding to image from Figure 1.g and its complementary. These images, shown in Fig. 2, are used to test the segmentation procedure on the basis of the local textural properties. We can notice that the resulting composite images show some very small areas with different textures, as compared to the size of the primary grains, for which the segmentation is a very difficult task.

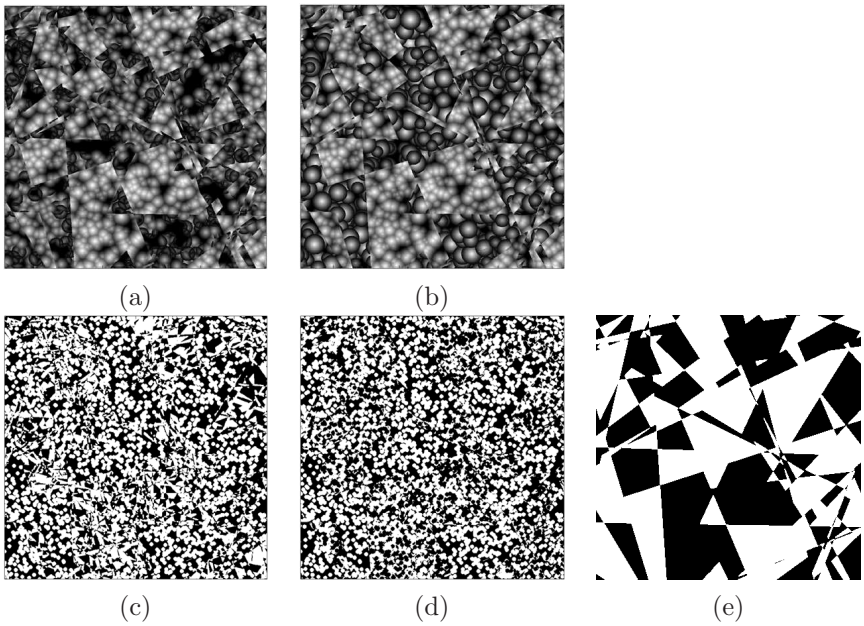


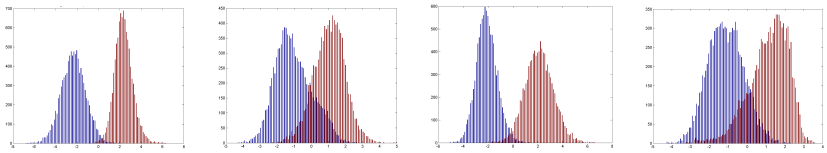
FIGURE 2. Patchwork image containing two types of texture for separation. Combination of: (a) Figure 1.a and b. (b) Figure 1.a and c. (c) Figure 1.d and e. (d) Figure 1.d and f. (e) The mask used for patchwork image: binary Poisson mosaic with 50 lines.

4.2 Experiments

Learning Procedure Texture descriptors are calculated both on original images and on the patchwork image. For each patchwork image, two

different learning training procedure based on a linear discriminant analysis (LDA) are produced and compared. In each case, both training and testing pixels are extracted. First, we randomly extract from each of the two original images (Fig. 1), 20,000 pixels half for the training and the rest for the test. This is labeled "Training ORI" in the following. This analysis aims to calculate the projection of descriptors that corresponds to the best linear separation between the two textures. Second, we extract from half of the patchwork image (Fig. 2), 10,000 pixels of each texture for the training and from the other half of the patchwork image 10,000 pixels of each texture for the test. This is labeled "Training PATCHW" in the following. This analysis aims to calculate the projection of descriptors that allows to best discriminate between the two textures even in border areas. We have to stress the fact that the textures in the original images (Figure 1) and in the patchwork images (Figure 2) have different probabilistic properties, since the second ones are a combination of simple textures and of a random set (the large Poisson mosaic).

An histogram of train data projections on the first LDA axis is presented in figure 3.



(a1) (a2) (b1) (b2)

FIGURE 3. Exemple of histogram of training data projection on the first LDA axis when training (1) ORI, (2) PATCHW. (a) For gray scale image corresponding to Fig. 2.a. (no averaging of the descriptors) (b) For binary image corresponding to Fig. 2.c (descriptors are averaged by a box of size 35×35).

It shows that the separation between the two textures is systematically reachable. The separation appears to be more efficient for training ORI than training PATCHW. In this last case, the descriptors for one texture is clearly influenced by the descriptors from the other ones, in particular for pixels located near boundaries. It confirms that textures have different probabilistic properties. The Gaussian shape suggests that LDA was indeed a good method, much simpler to perform than SVM (however we provide a comparison of the results obtained for the two techniques). It leads us to model the data as a mixture of two Gaussian random variables.

Finally, the patchwork image descriptors are projected on the first LDA axis and classified. The probability for each pixel of belonging to one of the Gaussian distribution is calculated and is used to produce the classification.

Using the knowledge of the mask, a patchwork error corresponding to a misclassification is evaluated. It is the global error of the approach.

The difference between train and test errors are smaller than 1.5 % in all the cases, showing that the system do not overfit. Therefore we restrict the presentation to test errors.

Different sizes of windows B are used to average descriptors as detailed in section 3.3. The figure 4 presents the test errors and the patchwork errors as a function of this size for the two learning procedures (training ORI and PATCHW).

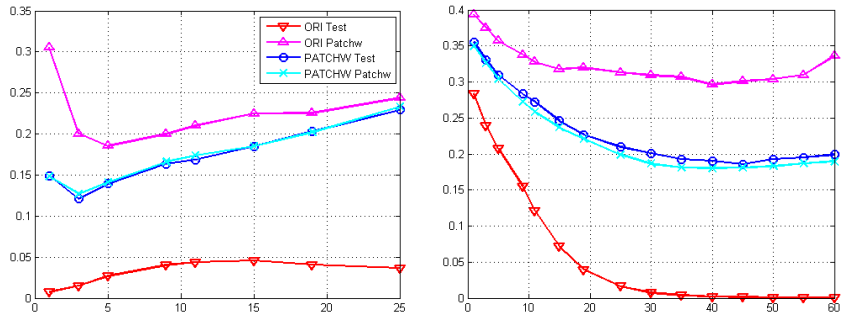


FIGURE 4. Errors versus size of descriptor averaging boxes (a) For gray scale image corresponding to Fig. 2.a. (b) For binary image corresponding to Fig. 2.d. ∇ test error for training ORI. $\triangle$ patchwork error for training ORI. O test error for training PATCHW. X patchwork error for training PATCHW.

Only two of the four projections are presented, because the results are similar for the two gray level images and for the two binary images. However, the behaviors are very different between gray scale and binary images used in this study.

Training on original vs. patchwork images The gap between test errors and patchwork errors for training ORI shows that this training procedure produces a large error, as a result of edge effects induced by the boundaries of domains with a single texture. Training on images with a single texture is therefore not very efficient, and should be avoided for applications to texture segmentation when they are intimately mixed.

Influence of the window size of the post-averaging (a) For gray scale images, only a small box of size 3×3 slightly improves the results. It shows that the descriptors on the scale of pixels are well adapted to our problem in the case of the gray level images.

(b) For the training on original images, we note that increasing the averaging window size systematically decreases the test error: the separation of the textures increases, as a result of a better estimate of the descriptors. However, the precision of the detection decreases, in particular near the boundaries. Therefore we have to find a compromise, corresponding to the minimum reached on the patchwork error curve.

In the following, we keep no averaging for the gray scale and a 35×35 averaging box for the binary images. A summary of the test and patchwork errors is given in table 1.

	Training ORI				Training PATCHW				
	No Average		Average		No Average		Average		SVM
Fig.	Test	Patch	Test	Patch	Test	Patch	Test	Patch	Patch
2.a	1.0	29.5	NA	NA	15.5	**15.7**	NA	NA	9.7
2.b	0.8	30.5	NA	NA	14.9	**14.8**	NA	NA	9.4
2.c	25.6	30.6	1.8	15.8	29.1	29.2	13.2	**13.5**	12.1
2.d	28.4	39.4	3.7	30.7	35.6	35.0	19.3	**18.1**	19.3

TABLE 1. Test and patchwork errors for the different proposed procedures. Average box sizes 35×35. All values are in %.

Results analyses With the present approach we obtain between 82 % and 90 % of pixels that are correctly classified. These results are very satisfying taking into account the complexity of the chosen images. We run a SVM using a radial basis function ($\exp(-\frac{1}{2}(u-v)^2)$) for the best training procedure (train PATCHW with the adapted averaging window size). By minimizing the test error, we optimized the regularization parameter C that controls the trade-off between training set accuracy and generalization performance. The result, presented in table 1 are similar to LDA ones for the binary images and greatly improved for gray scale images.

Chosen descriptors We have also looked for descriptors giving a larger contribution to the LDA projection. The disk structuring element appears to be the most relevant one for all present applications, where the textures are isotropic. The erosion/ dilation are most relevant for binary images and opening/closing for the gray level images. The present curvelet approach, which acts symmetrically on images and on their negative, is unable to separate textures like the sequential alternate and the Boolean random function. Indeed, the curvelets are more relevant for binary images than grey level images.

The typical scale of the present binary images is very small (within a range of 12 pixels), and therefore the contribution of structuring elements of large size decreases, whatever the considered training.

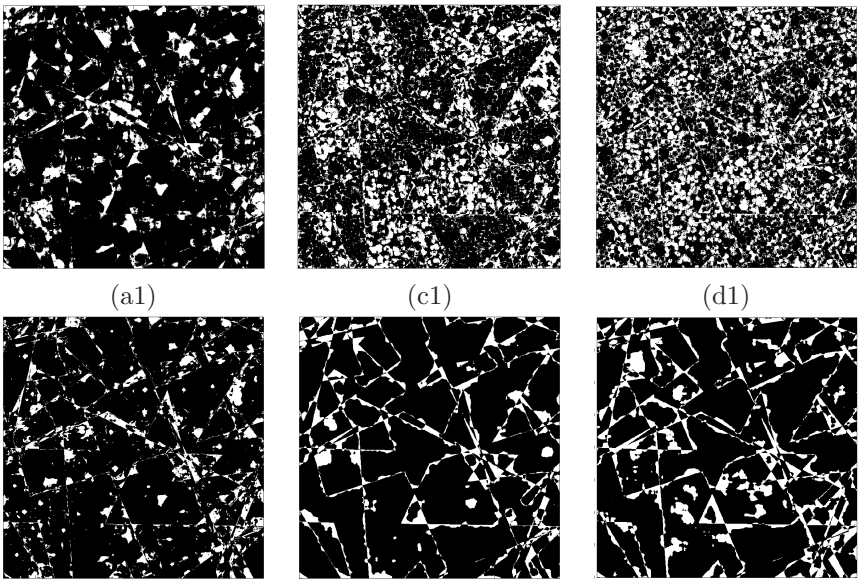
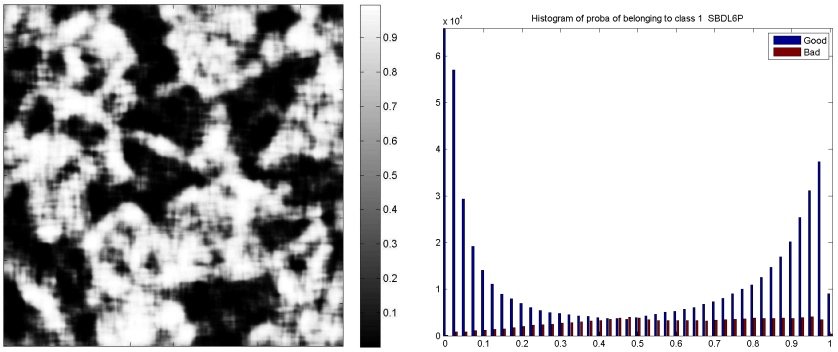


FIGURE 5. Localisation of misclassified pixels are plotted in white for the 4 studied cases. a, b , c and d correspond to the label in Figure 2. (1) Descriptors are not averaged. (2) Descriptors are averaged by a 35×35 box.

Error localization The localization of misclassified pixels gives interesting information. For gray scale images (Fig. 5.a and 5.b) errors are mainly located in small areas of the patchwork image and near the boundaries between the two textures. For binary images, this Figure illustrates the differences existing between non averaged descriptors (Fig. 5.c1 and 5.d1) and averaged descriptors (Fig. 5.c2 and 5.d2). The improvement of the descriptors averaging is clearly visible: for the training 1 large areas of identical texture are misclassified, and the error image appears very noisy. On the contrary, for the training 2, the localization of misclassified pixels are mainly located on the boundary between the two textures.

We plot in figure 6 the classification probability for image 2.d and the corresponding histogram. The probability image could be used as a seed for further segmentation, using for instance watersheds. However, our attempts and a comparison to image 2.d shows that it would be very difficult to recover the misclassified pixels in small area without introducing errors in the well classified pixels. Therefore the present classification is closed to the optimal one.



(a) (b)

FIGURE 6. (a) Classification probability for image 2.d. (b) The corresponding histogram. In read pixels that are misclassified and in blue the well classified ones.

5. Conclusion

A morphological approach at the level of pixels in images, combined with statistical learning proved to be very efficient for the segmentation of binary or digital textures, as illustrated for difficult case studies. The main conclusions for further applications are as follows: the basic operations of mathematical morphology (erosion/dilation for binary images, and opening/closing for gray level images) provide efficient descriptors of textures. Local averaging of the binary descriptors improves significantly their discriminant power. Finally, the training procedure has to be made on a selection of pixels in composite images, rather than in pure textures.

Bibliography

[1] A. Aubert, D. Jeulin, and R. Hashimoto, *Surface texture classification from morphological transformations*, Proc. ISMM'2000, Mathematical morphology and its applications to Image and Signal Processing, 2000, pp. 253–252.

[2] A. Aubert and D. Jeulin, *Classification morphologique de surfaces rugueuses*, Revue de Métallurgie - CIT/Sience et Génie de Matériaux **Feb 2000** (2000), 253–262.

[3] E. J. Candes and D. L. Donoho, *Curvelets - a surprisingly effective non-adaptive representation for objects with edges*, A Cohen,C Rabut,L L Schumaker Eds. Curve and Surface Fitting, 1999.

[4] F.S. Cohen, Z. Fan, and S. Attali, *Automated inspection of textile fabrics using textural models*, IEEE Transactions on Pattern Analysis and Machine Intelligence **13** (1991), no. 8, 803–808.

[5] M. Elad, J.L. Starck, D. Donoho, and P. Querre, *Simultaneous cartoon and texture image inpainting using morphological component analysis (mca)*, ACHA (to appear).

[6] G. Fricout and D. Jeulin, *Analisis de imgen y morfologia matematica para la caracterizacion en linea del aspecto de las superficies pintadas*, Pinturas y Acabados **XLVI** (2004), no. 292, 6–13.

[7] J. Iivarinen, *Surface defect detection with histogram-based texture features*, Intelligent robots and computer vision xix: Algorithms, techniques, and active vision, 2000, pp. 140–145.

[8] D. Jeulin, *Multivariate random image models*, Acta Stereologica; 11/SUPPL I (1992), 59–66.

[9] ———, *Random texture models for materials structures*, Statistics and Computing **10** (2000), 121–131.

[10] D. A. Karras and B. G. Mertzios, *Improved defect detection using novel wavelet feature extraction involving principal component analysis and neural network techniques*, Australian joint conference on artificial intelligence, 2002, pp. 638–647.

[11] A. Kumar and G.K.H. Pang, *Defect detection in textured materials using gabor filters*, Industry Applications **38** (2002), no. 2, 425–440.

[12] G. Lambert and F. Bock, *Wavelet methods for texture defect detection*, Proceedings of the 1997 international conference on image processing, 1997, pp. 201–204.

[13] J. Malik, S. Belongie, J. Shi, and T. Leung, *Textons, contours and regions: Cue integration in image segmentation*, Seventh international conference on computer vision (iccv'99), 1999, pp. vol. 2, p. 918.

[14] G. Matheron, *Random sets and integral geometry*, J. Wiley, 1975.

[15] A. Serdaroglu, A. Ertuzun, and A. Ercil, *Defect detection in textile fabric images using wavelet transforms and independent component analysis*, Journal Pattern Recognition and Image Analysis **16** (2006), no. 1, 61–64.

[16] J. Serra, *Image analysis and mathematical morphology*, Academic Press, London, 1982.

[17] K. Sivakumar and J. Goutsias, *Discrete morphological size distributions and densities: estimation techniques and applications*, Journal of Electronic Imaging **6** (1997), no. 1, 31–53.

[18] X. Xie and M. Mirmehdi, *Texture exemplars for defect detection on random textures*, Icapr, 2005, pp. 404–413.