Mixed Texture Composition: Determining the Proportion of Structured and Statistical Components.

Carolyn Evans[1] and Han Zhang[2]

[1] CSIRO Mathematical and Information Sciences, Locked Bag 17, North Ryde 1670, NSW, Australia,
`carolyn.evans@csiro.au`
[2] Department of Mathematics, University of New South Wales, Australia

Abstract. How should one model a texture? Statistically? Structurally? Or a bit of both? In this paper, we propose a method to help one decide. We assume a texture contains a random and a structured component, then estimate the proportion of energy in each component using a new algorithm. The energy proportions indicate whether one texture component dominates or an even mix exists. We implemented our algorithm on known texture mixtures and obtained energy proportion estimates that were accurate to ±0.003. This compares well to raw estimates made using an established method. We also estimated the proportions of random and structured components in natural textures taken from a public database. We were able to order the 2D textures in a perceptually-reasonably way using the estimated energy proportion of the structured component. Unfortunately the new algorithm showed signs of being directionally biased and not invariant to non-uniform lighting conditions across a texture. Thus further development is required.

1 Introduction

Many algorithms which segment or classify image texture assume either a structural model or a statistical model of texture. The most suitable model depends on the texture itself. For example, the texture in Fig. 1a is highly structured in two dimensions. It could be described by a primitive unit (a hole surrounded by a local white region) regularly tiled in two dimensions using a certain set of rules. Julesz [10] first proposed this structured model of texture, referring to a texture's primitive unit as a 'texton'. The structured model of texture has been adopted in [15, 1, 2, 16].

The texture in Fig. 1b, however, cannot be broken down into texture primitives. It appears to be the result of a statistical process and would be better described using a statistical model. The works of [7, 12, 4] focus on these type of textures.

The texture in Fig. 1c presents a problem. It appears to be a mixture of a structured texture component and a statistical texture component. Many natural
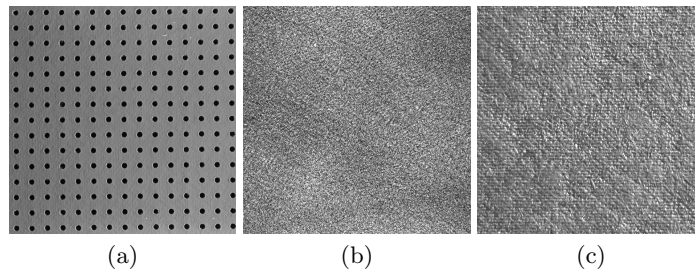
721

| (a) | (b) | (c) |

Fig. 1. (a) A highly deterministic texture. (b) A highly random texture. (c) A texture that contains a deterministic component and a random component. These textures are from the VisTex public database [14].

textures are like this. Visually it is difficult to tell which component dominates and therefore which texture model is more appropriate. May be neither component dominates and a mixed model is required, as proposed in [5, 8, 13].

In this paper we propose a method of objectively estimating the proportion of energy in the structured and statistical components of a texture. We apply out method to synthetic textures of known composition and find our results accurate to ± 0.003 or 0.3%. This was superior to the accuracy of estimates made using Liu and Picard's algorithm [13]. Our method is designed for textures that exist in two spatial dimensions, in greyscale (single band) images. The method can be easily generalised for textures that exist in two spatial dimensions but multi-band images, such as colour, by treating each image band independantly.

In Section 2 we explain how we estimate the proportion of energy in the structured and statistical texture components. We also explain how Liu and Picard estimate these proportions as part of their 'Harmonicity Test'. In Section 4 we compare the results of the two methods in detail when applied to synthetic textures of known composition. We also show the results of applying both methods to a range of natural textures from the VisTex database [14], where the composition is unknown. In Section 5 we conclude that our energy proportion estimates are a first step towards an objective measure of which texture model; structured, statistical or mixed, is the most suitable for a particular texture.

2   Estimating the Mixture Proportions of a Texture

2.1   Our Method

We summarise our method of estimating the proportion of energy in the structured and statistical components of a texture by;

1. Subtract the texture's mean grey value from the texture before computing its 2D Discrete Cosine Transform (DCT).

2. Determine the orientation and eccentricity of elliptical contours in the distribution of the texture's DCT coefficients.

Then along each elliptical contour, k, calculate

3. the number of datums, N_k,
4. the sample mean, μ_k of the DCT coefficients,
5. the sample variance σ_k^2 of the DCT coefficients
6. and the cumulative probability of each DCT coefficient assuming that $(DCT_{i,j} - \mu_k)^2/\sigma_k^2$ follows a χ^2 distribution with N_k degrees of freedom.
7. If the cumulative probability of a DCT coefficient is greater than a user-defined threshold, t, classify it as caused by the structured texture component.
8. Given that DCT_{STRUCT} are the set of coefficients classified as arising from the structured texture component, calculate

$$E_{STRUCT} = \sum_{i,j \in STRUCT} DCT_{i,j}^2, \tag{1}$$

$$E_{TOTAL} = \sum_i \sum_j DCT_{i,j}^2, \tag{2}$$

$$P_{STRUCT} = \frac{E_{STRUCT}}{E_{TOTAL}}, \tag{3}$$

$$P_{STAT} = 1 - P_{STRUCT}. \tag{4}$$

In step 1 of the algorithm, we subtract the texture's mean greyvalue before computing the DCT because the global mean does not contain any texture information. From the work of [11], we expect that the DCT coefficients of the statistical texture component follow a 2D Laplacian distribution. However we don't assume this. We do assume that the 2D distribution of DCT coefficients,

– is centred at $(i,j) = (0,0)$,
– is non-separable in the orthogonal spatial dimensions, i and j
– and its domain can be divided up into concentric partial ellipses (contours). The DCT coefficients on a particular elliptical contour are assumed to come from a particular Normal distribution.

In step 2 of our method we determine the orientation and eccentricity of the elliptical contours using Principal Component Analysis (PCA) [6]. This means we calculate the covariance of the distribution of DCT^2 coefficients in the two dimensions i, j with respect to the origin $(0,0)$. The covariance matrix is made up of the elements $C_{i,i}, C_{j,j}, C_{j,i}$ and $C_{i,j}$ given by

$$C_{i,i} = \frac{\sum_i \sum_j (DCT_{i,j}^2 i^2)}{\sum_i \sum_j DCT_{i,j}^2}, \quad C_{j,j} = \frac{\sum_i \sum_j (DCT_{i,j}^2 j^2)}{\sum_i \sum_j DCT_{i,j}^2},$$
$$and \ C_{i,j} = C_{j,i} = \frac{\sum_i \sum_j (DCT_{i,j}^2 ij)}{\sum_i \sum_j DCT_{i,j}^2}. \tag{5}$$

The first and second eigenvectors of this matrix are parallel to the major and minor axes respectively of the elliptical contours. The orientation of the first eigenvector relative to the positive i direction defines the orientation of the elliptical contours. The ratio of the two eigenvalues defines the eccentricity of the contours.

Underlying steps 3 - 7 of our method is our assumption that each DCT coefficient arising from the statistical texture component follows a Normal distribution, $DCT_{i,j} \approx N(\mu_{i,j}, \sigma_{i,j}^2)$. We expect the means of these Normal distributions to be the same for DCT coefficients located on the same contour. We estimate this mean from the sample mean, μ_k, for each contour, k, in step 3.

The variances, $\sigma_{i,j}^2$, of DCT coefficients on the same elliptical contour are not necessarily the same. If we had multiple images containing textures generated by the same stochastic process, we could calculate the sample variance for each value of (i, j), to estimate each $\sigma_{i,j}^2$. However, we have only one image of each texture. Due to this limitation we cannot estimate each sample variance $\sigma_{i,j}^2$ assuming each is independent of the others. For any progress to be made, we must assume that the variance $\sigma_{i,j}^2$ for each DCT coefficient on the same contour is the same. We estimate it in step 4 as the sample variance along each contour, σ_k^2.

Assuming that DCT coefficients on elliptical contour k are $N(\mu_k, \sigma_k^2)$ distributed, the quantity $(DCT_{i,j} - \mu_k)^2/\sigma_k^2$ should follow a χ^2 distribution with N_k degrees of freedom [9]. If the quantity $(DCT_{i,j} - \mu_k)^2/\sigma_k^2$ is located in the extreme tail of this distribution, it is atypical of the distribution. This will be reflected by its cumulative probability being very high. In step 7 of our algorithm we threshold the cumulative probability at the user-defined value $t \in [0, 1]$. We presume that DCT coefficients that have a greater cumulative probability than t arise from the structured texture component. Those that have a cumulative probability below the threshold fit our model of the statistical texture component well. The results of this thresholding step do depend on the chosen value of t. We therefore applied the algorithm repeatedly using a range of t values for each texture (Section 4) to find out the extent of parameter t's influence.

Once the DCT coefficients arising from the structured texture component have been identified, we calculate the proportion of energy in the structured and statistical texture components using Eqs. (1)-(4).

2.2 Liu and Picard's Method

Liu and Picard [13] proposed that a texture could be decomposed into harmonic (2D periodic), evanescent (1D periodic) and random components in the Fourier domain. This was conditional on a texture passing a 'Harmonicity Test'. The harmonicity test is based on the ratio of energy at small displacements, r_e, in the texture's auto-covariance function. From the spectral properties of time series [3] it is known that random textures should have relatively high proportions of energy at small displacements. Highly structured textures, however, should have energy concentrated periodically through its auto-covariance function.

Liu and Picard calculate the ratio, r_e by;

1. multiplying the texture image by a 2D Gaussian taper,
2. calculating the 2D auto-covariance function (ACF) of the tapered image,
3. thresholding the ACF at a percentage of the total image energy (threshold specified by the user),
4. calculating the percentage of the total energy in the ACF which is connected to the origin and is greater than the threshold.

The value of the ACF at the origin is ignored in this calculation since the texture is independent of the image mean. Further details of the algorithm can be found in [13].

We implement this algorithm in Section 4 using Liu and Picard's code, publicly available from **ftp://whitechapel.media.mit.edu** in directory **/pub/fliu/wold/**. Although Liu and Picard did not base their harmonicity test directly on their raw estimates of r_e $(1 - r_e)$, we do directly compare the values with our estimates of P_{STAT} (P_{STRUCT}) respectively. Our reason was that these quantities were the most comparable quantities we found in the literature.

3 Test Images

To test the accuracy of our method, we generated a set of synthetic textures of known composition. Each texture in the set was formed from a linear combination of either the purely random texture of Fig. 2a or Fig. 2b, and the purely structured texture of Fig. 2c.
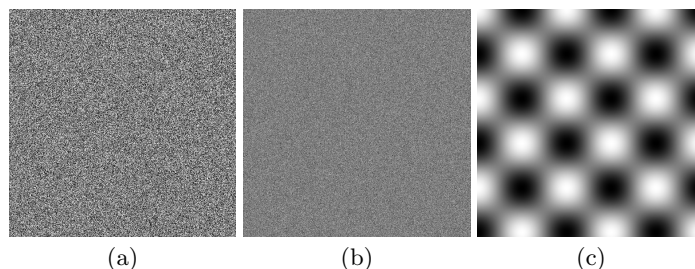

(a)   (b)   (c)

Fig. 2. Purely random textures containing iid pixel values from (a) a uniform distribution and (b) a Normal distribution. (c) Purely structured texture.

Each of the images in Fig. 2 are 8-bit and 512×512 pixels. The pixel values in each random texture are independently and identically distributed. The pixel values in Fig. 2a, however, are from a uniform distribution with zero mean, whereas the pixel values in Fig. 2b follow a Normal distribution with zero mean. The structured texture is a 2D cosine function with zero mean.

We also tested the performance of the two methods on greyscale version of Fabric and Tile textures taken from the public VisTex database [14]. In this

database, each image of a fabric, food or tile texture is referred to by a unique number within that texture type, eg. Fabric0000-Fabric0019. We refer to the texture images using the same names and numbers in this paper, which explains the choice of labels on the horizontal axes of Figs. 5a-d.

The VisTex textures were also 512×512 pixels in size but were not artificially generated. Thus, we don't know the actual proportions of energy in the structured and statistical components of these textures. The results of our algorithm must be subjectively evaluated by eye for these cases.

4 Results

4.1 Synthetic Texture Results

Figure 3a and b show the error in our estimates of the proportion of energy in the structured components of the set of synthetic mixed textures. The error is plotted on the vertical axis and the known energy proportions are plotted on the horizontal axis (which is not to scale). We have used box plots to show the range of error obtained when variable t in step 7 of our algorithm was varied from $0.5 \leq t \leq 0.999$.
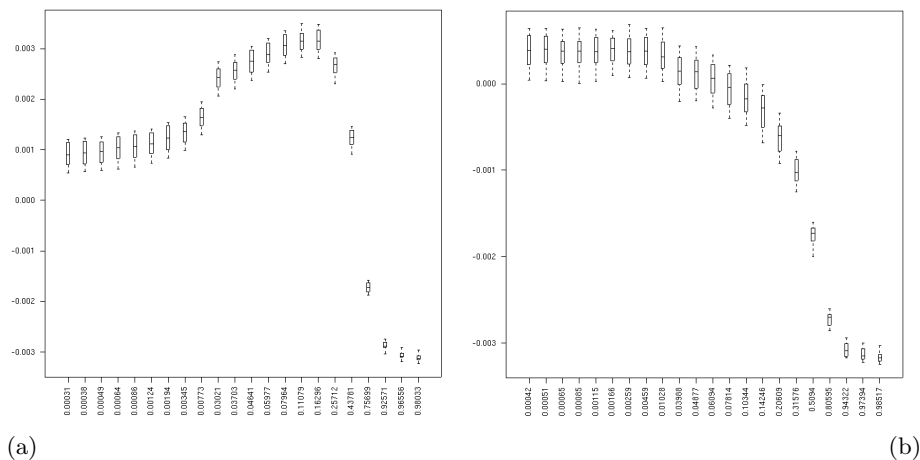

(a)   (b)

Fig. 3. Error in our estimates of P_{STRUCT} (vertical axis) plotted against actual P_{STRUCT} (horizontal axis) for various mixtures of (a) textures Fig. 2a and c, and (b) textures Fig. 2b and c.

Similarly Figure 4 shows the error in using Liu and Picard's quantity of $(1 - r_e)$ to estimate the proportion of energy in the structured components of the set of synthetic mixed textures. We again use box plots to show the range of

error because we varied the percentage threshold in step 3 of their method from 10% down to 0.01%.
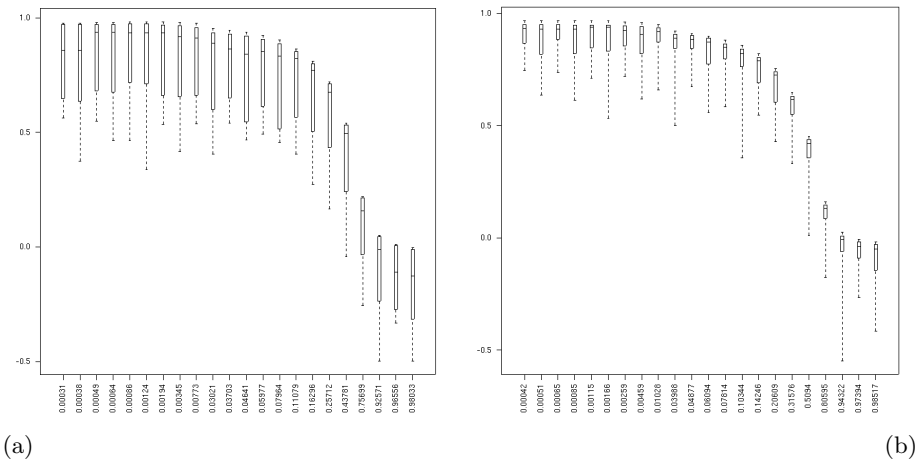

(a)   (b)

Fig. 4. Error in using Liu and Picard's quantity $(1 - r_e)$ to estimate P_{STRUCT} for various mixtures of (a) textures Fig. 2a and c, and (b) textures Fig. 2b and c.

By comparing Fig. 3 with Fig. 4, we find our estimates of P_{STRUCT} are more accurate than Liu and Picard's quantity $(1 - r_e)$. Additionally our estimates are more reliable. This is indicated by narrower box plots in Fig. 3 than Fig. 4.

Currently our method is run as a script under 'R'. It takes longer to run than Liu and Picard's method, but their's has been implemented in 'C'. We therefore only compare the accuracy of the two methods and not the computational speed.

4.2   Fabric and Tile Texture Results

The results of applying both methods to Fabric and Tile textures are shown in Fig. 5. Again, we applied both methods repeatedly using different threshold values. In the case of our method, 22 different thresholds were used in the range $0.5 \leq t \leq 0.999$. In the case of Liu and Picard's method, 20 thresholds in the range $0.01\% \leq t \leq 10\%$ were used. The box plots show the range of P_{STRUCT} estimates obtained for each texture as a result. The narrower box plots in Figs. 5a and c indicate that our method is less sensitive to the chosen threshold value than Liu and Picard's method.

The centre markings of the box plots in Figs. 5a-d indicate that our estimates of P_{STRUCT} respond more sensibly and consistently to the structured component of the natural textures than the quantity $(1 - r_e)$. According to our P_{STRUCT} estimates Tile texture 9 (Fig. 5h) has a greater proportion of structure
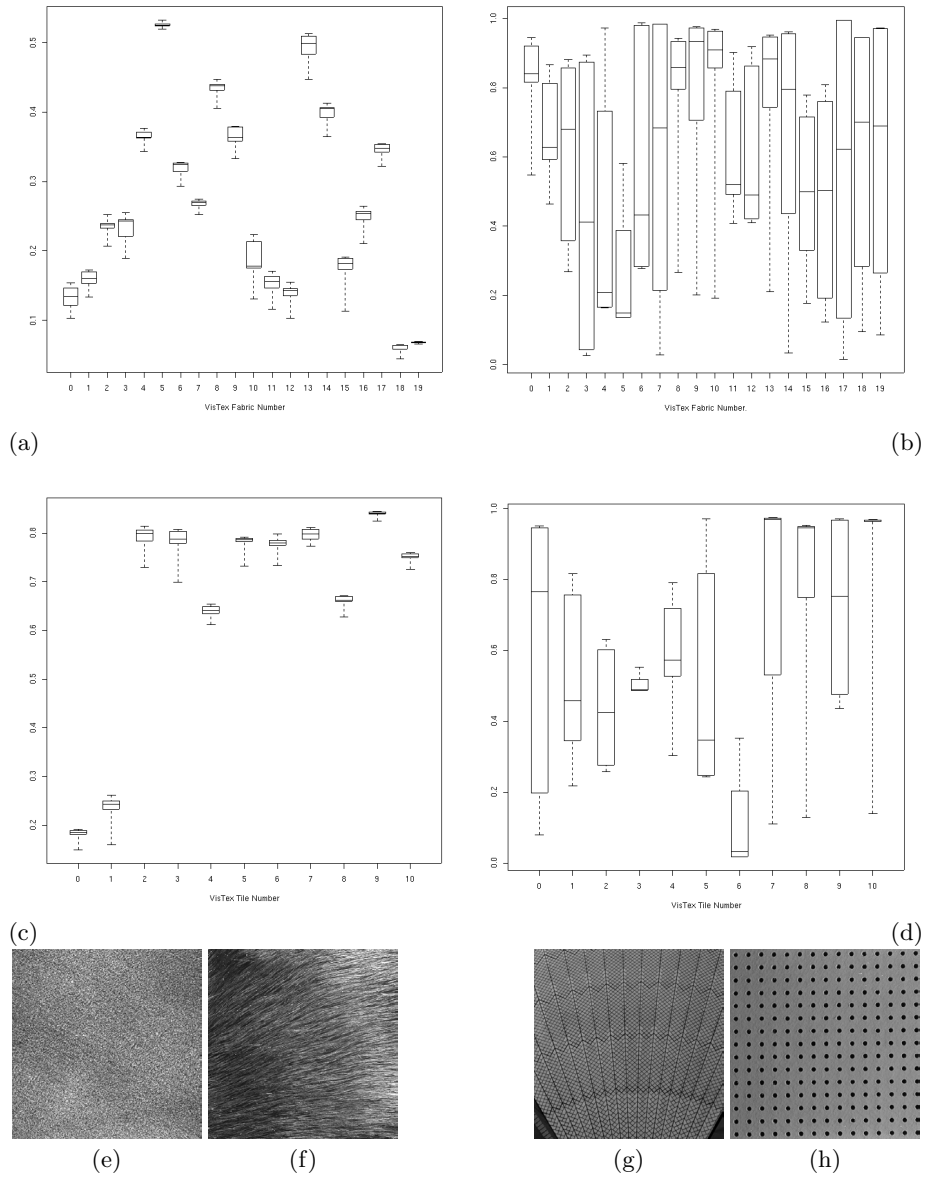
(a)

(b)

(c)

(d)

(e)

(f)

(g)

(h)

Fig. 5. (a) Our estimates of P_{STRUCT} for Fabric textures. (b) P_{STRUCT} estimates for Fabric textures based on the quantity $(1 - r_e)$. (c) Our estimates of P_{STRUCT} for Tile textures. (d) P_{STRUCT} estimates for Tile textures based on the quantity $(1 - r_e)$. (e) Fabric texture 18. (f) Fabric texture 5. (g) Tile texture 0. (h) Tile texture 10.

728

than Fabric texture 5 (Fig. 5f), which in turn has more structure than Fabric texture 18 (Fig. 5e). From visual inspection, this seems a reasonable result. In contrast, the quantity $(1 - r_e)$ suggests that Fabric texture 18 has around the same proportion of structure as Tile texture 9!

However, our P_{STRUCT} estimates for the natural textures are not perfect. Fabric texture 4, 5 and 6 all contain the same fabric, viewed from the same orientation but imaged under different, non-uniform lighting conditions. We would expect that our estimates of P_{STRUCT} should be very similar for these three textures, but Fig. 5a shows that this is not the case. Our estimation method is sensitive to these lighting changes. The difference in P_{STRUCT} for fabric texture 0 and 2 are also caused by changed lighting conditions.

Additionally our estimation method is directionally biased. This is evident from our P_{STRUCT} estimates for Tile texture 9 and 10. Tile texture 10 is a rotated version of Tile texture 9, yet our P_{STRUCT} estimate for Tile 10 is noticeably smaller than for Tile 9. The directional bias also explains the difference in P_{STRUCT} estimates for Fabric 0 and 1, Fabric 2 and 3, and partly explains the low P_{STRUCT} estimate for Tile texture 0.

These two weakness in our estimation algorithm originate from the global DCT used in step 1 of the algorithm. The global nature of the transformation means that it is not invariant to non-uniform lighting changes across a texture. The DCT is directionally biased because it extends all texture images by reflection at the horizontal and vertical image boundaries. This is a reasonably way of extending textures like Tile 9 with principal directions in the horizontal and vertical directions, but not rotated versions of them. We may be able to remove these two weaknesses from our algorithm by replacing the global DCT with a local wavelet transform, but further investigation is required.

5   Conclusion

We have presented a new method of estimating the proportion of energy in the structured and statistical components of a two-dimensional image texture. The results yielded were accurate to $\pm 0.3\%$ for synthetically generated textures. Based on visual inspection, our method estimated the structured component of natural textures more accurately and reliably than Liu and Picard's quantity $(1 - r_e)$ as well. Unfortunately the new algorithm does require further improvements so to remove its directional bias and make it invariant to non-uniform lighting conditions across a texture. This may be possible by replacing the use of a global DCT in the algorithm with a local wavelet transform.

References

1. Asano, A., Ohkubo, T., Muneyasu, M., Hinamoto, T.: Texture Primitive Description Using Skeleton. Proceedings of the International Symposium on Mathematical Morphology 2002, 101–107.

2. Asano, A., Endo, J.: Multiprimitive Texture Analysis Using Cluster Analysis and Size Density Function. Proceedings of the International Symposium on Mathematical Morphology 2002, 109–116.
3. Cryer, I.D.: Time Series Analysis. PWS Publishers, 1996.
4. De Souza, P.: Texture Recognition via Autoregression. Pattern Recognition, **15**(6), 471–475, 1982.
5. Francos, J.M., Meiri, A.Z., Porat, B.: A Unified Texture Model Based on a 2D Wold-Like Decomposition. IEEE Transactions on Signal Processing, **41**(8), 1993, 2665–2676.
6. Gnandesikan, R: Methods for Statistical Data Analysis of Multivariate Observations. 2nd Ed. John Wiley and Sons, Inc. 1997.
7. Hofmann, T., Puzicha, J., Buhmann, J.M.: Unsupervised Texture Segmentation in a Deterministic Annealing Framework. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1998.
8. Hsu, T-I., Wilson, R.: A Two Component Model of Texture for Analysis and Synthesis. IEEE Transactions on Image Processing, **7**(10), 1998, 1466–1476.
9. Johnson, N.I., Kotz, S: Continuous Univariate Distributions-1. p75. John Wiley and Sons, Inc. 1970.
10. Julesz, B.: Textons, the elements of texture perception and their interactions. Nature, **290**, March 1981, 91–97.
11. Lam, E.Y., Goodman, J.W.: A Mathematical Analysis of the DCT Coefficient Distribution for Images. IEEE Transactions on Image Processing, **9**(10), October 2000, 1661–1666.
12. Lee, T. C. M., Berman, M.: Nonparametric estimation and simulation of two–dimensional Gaussian image textures. Graphical Models and Image Processing 59, 434–445, 1997.
13. Liu, F., Picard, R.W.: Periodicity, Directionality and Randomness: Wold Features for Image Modelling and Retrieval., IEEE Transactions on Pattern Analysis and Machine Intelligence, **18**(7), July 1996, 722–733.
14. Picard, R., Graczyk, C., Mann, S., Wahman, J., Picard, L., Campbell, L.: Vision Texture Database. **http://www-white.media.mit.edu/vismod/imagery/VisionTexture/vistex.html**, Massachusetts Institute of Technology, USA, 1995.
15. Soille, P.: Morphological Texture Analysis: An Introduction. In Lecture Notes in Physics **600**, 215–237. Arch. Rat. Mech. Anal. **78** (1982) 315–333
16. Voorhees, H., Poggio, T.: Computing texture boundaries from images. Nature, **333**, May 1988, 364–367.