BMVC99

Texture Deconvolution for the Fourier-Based Analysis of Non-Rectangular Regions

A. A. Clark[†], B. T. Thomas[†], N. W. Campbell[†], P. Greenway[§]
[†]Advanced Computing Research Centre, University of Bristol,
Bristol, BS8 1UB, UK.
[§]British Aerospace PLC, P.O. Box 5, Filton,
Bristol, BS12 7QW, UK.
Angus.Clark@bristol.ac.uk

Abstract

Fourier analysis is often used as a tool to facilitate the extraction of texture information from image data. However, in situations where the texture patch does not entirely fill the region of analysis, information relating to the shape of the patch becomes entwined with its texture content, thus contaminating the Fourier spectrum and corrupting the texture information. We propose the use of a frequency deconvolution algorithm to remove the artefacts introduced by shape components, permitting the Fourier-based analysis of non-rectangular image patches. The algorithm is demostrated on a texture recognition task involving the entire Brodatz album.

1 Introduction

Texture is an important source of information for a number of computer vision and graphics applications. Although it is difficult to construct a formal definition, texture is intuitively related to luminance variations in the image and can be characterised using properties such as regularity, coarseness, contrast, local and global structure, and directionality [4, 14]. A system that is required to deliver meaningful judgements concerning texture, needs to be able to extract a description of the image data in a form that explicitly captures these properties.

It has been suggested that the Fourier domain may provide a more favourable environment in which to work and a number of extensions, aimed at exploiting the Fourier representation of texture information, have been proposed [1, 6, 8]. However, many of these techniques have been demonstrated using tasks involving only rectangular texture patches. In real-world environments, we are confronted by a wide range of different object shapes with further variation introduced by the effects of occlusion. Due to the nature of the Fourier analysis, if the texture patch we wish to analyse does not entirely fill the region of analysis, then information relating to the shape of the patch becomes entwined with its texture content thus contaminating the Fourier spectrum and corrupting the texture information.

In this study, we investigate how the shape of the patch effects its Fourier representation, and what the repercussions might be for the overall analysis of the texture. We go on to propose the use of a frequency deconvolution algorithm to remove these effects and

BMVC 1999 doi:10.5244/C.13.20

provide a shape-invariant Fourier description of the texture content of non-rectangular image patches. The approach is based on the CLEAN deconvolution algorithm [5], originally developed for aperture synthesis in radio astronomy. We adapt the algorithm, and show how it can be applied to the analysis of non-rectangular texture patches.

2 Texture Recognition

In this section a texture recognition system is described. The decision process is driven by features derived from the principal components analysis of the Fourier domain [8]. The system is used here to investigate the effects of shape variation on Fourier-based texture analysis and to demonstrate the ability of the deconvolution algorithm to remove these effects.

2.1 The Texture Database

The texture patches used in this study were taken from the Brodatz album [2]. The album consists 112 pictures of natural textures and represents a diverse range of texture properties. Each picture in the album was scanned to produce a 640×640, 8-bit grey level image. From each of these source images, nine non-overlapping 128×128 sub-regions were extracted to form a set of 1008 square texture patches. A similar set of non-rectangular patches was also constructed. Each patch here was extracted using a mask chosen at random from a set of 10 possible shapes. As before, nine non-overlapping sub-regions were extracted from each source image. To insure a fair comparison between the two sets, the size of each shape mask was controlled such that the same surface area of texture was revealed as that of the square patches. Examples of the texture patches, taken from both sets, are shown in Figure 1.

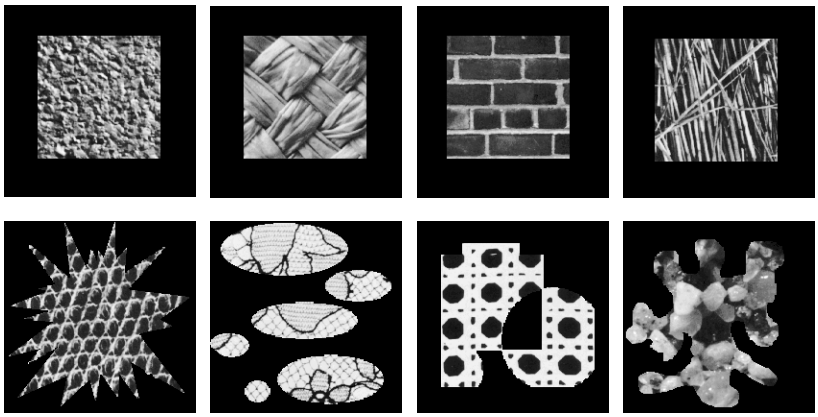

Figure 1: Examples from the Texture Database: square patches (top); non-rectangular patches (bottom).

2.2 Texture Features

The texture features are derived from the principal components analysis (PCA) of the Fourier domain as described in [8]. For the purpose of completeness, we summarise the method here.

From the set of square texture patches, ten percent ($p = 100$) are chosen at random to provide a "training" set for the analysis. For each image in the training set, the Discrete Fourier Transform (DFT) is computed and a 1-d vector, representing the magnitude components in raster-scan order, is formed: $\mathbf{x}_i \in \mathbf{R}^{n\times 1}, i = 1,\ldots,p : n = 128^2, p = 100$. To minimise the effects caused by boundary discontinuities, each image was mixed with a Gaussian window ($\sigma = 24$) prior to the transform. The covariance matrix is calculated from: $\mathbf{C} = \frac{1}{p}\sum_{i=1}^{p}(\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})' = \mathbf{X}\mathbf{X}'$ where $\mathbf{m}$ is the sample mean, $\mathbf{m} = \frac{1}{p}\sum_{i=1}^{p}\mathbf{x}_i$. The eigenvectors, $\mathbf{q}_i$, and corresponding eigenvalues, λ_i, are determined by the characteristic equation:

$$\mathbf{C}\mathbf{q}_j = \mathbf{X}\mathbf{X}'\mathbf{q}_j = \lambda_j \tag{1}$$

Using the fact that there can be at most p eigenvectors and $p \ll n$, computation can be saved by first performing the eigen analysis on the inner product:

$$\mathbf{X}'\mathbf{X}\mathbf{u}_j = \lambda_j\mathbf{u}_j \tag{2}$$

Pre-multiplying both sides of 2 by $\mathbf{X}$ gives:

$$\mathbf{X}\mathbf{X}'(\mathbf{X}\mathbf{u}_j) = \lambda_j(\mathbf{X}\mathbf{u}_j) \tag{3}$$

From Equation 3, it is observed that the eigenvectors for the original covariance matrix, $\mathbf{X}\mathbf{X}'$, are given by $\mathbf{q}_j = \mathbf{X}\mathbf{u}_j$.

The eigenvectors represent an alternative orthogonal basis whereby the importance of each axis, in terms of the variance it accounts for, is given by its corresponding eigenvalue. The vectors, ordered by their eigenvalues, reveal the principal modes of variation.

The texture features are formed by projecting the DFT magnitudes onto the principal modes. In the experiments reported here, the first 40 modes were used as texture features. Accounting for over 85% of the variance found in the training set, this choice provides a compact, yet expressive representation of power spectra typical to those generated by textures.

3 Shape Variation on Texture Analysis

3.1 Experiments

A recognition task is set up to determine the effect of introducing arbitrarily-shaped image patches on Fourier-based texture analysis. For a given patch, the system is required to decide which of the 112 Brodatz texture it most closely corresponds to. The classification is performed using a k-nearest-neighbour paradigm, based on the Euclidean distance between feature vectors. For each of the 112 textures, 4 member patches are used to form an estimate of the class distributions over the feature space. These patches are excluded from the rest of the analysis. In the experiments reported here, k was set to 3.

The system was applied to both sets of texture patches in turn.

3.2 Results and Discussion

The overall classification accuracy attained for each set is shown in Figure 2, with individual results given for a selection of texture classes. For the set of square texture patches, over 76% were correctly classified, achieving a level of performance similar to those reported by other Fourier-based techniques. The fact that a relatively simple classification paradigm was used, suggests that the feature vector is largely responsible for the performance of the system. On closer inspection of the misclassifications, further evidence is revealed supporting the ability of the feature vector to capture perceptual concepts of texture. For example, although one D022 patch was misclassified as D003, both are examples of snake skin. It is, in fact, reassuring to find that these two perceptually similar textures generate measurably close feature vectors, despite the confusion and misclassification that arises. Further incidents of misclassification were found for patches belonging

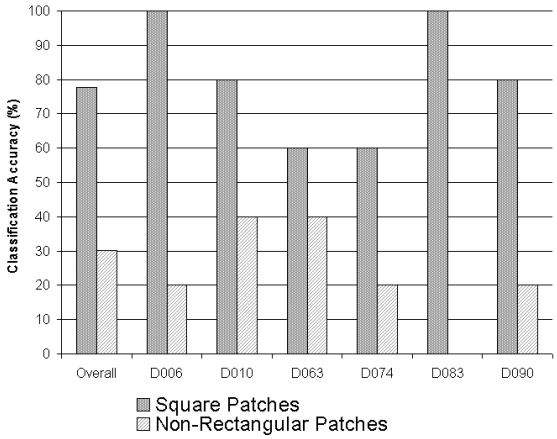

Figure 2: Effects of shape variation on texture recognition

to large scale textures. It was observed that in theses case, the underlying pattern often spans the entire source image, and as a result, each extracted patch only contains part of the texture. Given these circumstances, it is unreasonable to expect the system to deliver an accurate classification when only partial information is available. Indeed, it is perhaps undesirable to match these patches which appear so unrelated, even if they do originate from the same source.

As an overall observation, it was found that regular textures enjoyed a slight advantage in classification accuracy.

Turning to the results found for the set of non-rectangular texture patches, it is clear that introducing shape variations can severely disturb the analysis of texture patches. The overall classification dropped by over 45% to only 30% correctly classified, with deterioration in accuracy noted across almost all classes.

The textures feature have shown to be vulnerable to the artefacts introduced by the shape components, responding to properties relating to the shape of the patch instead of its texture content. Although we have used PCA to extract texture features from the

Fourier domain, it is likely that other Fourier-based techniques would suffer a similar fate when confronted by arbitrarily-shaped regions.

4 Frequency Deconvolution

4.1 Motivation

The experiment conducted in the previous section provides a vivid demonstration of the detrimental effects that shape variations can have on the analysis of texture patches. If Fourier-based techniques are to deliver meaningful judgements concerning the texture content of non-rectangular regions, then a spectral representation of the texture needs to be extracted that is invariant to the shape of the patch. This remains a challenging undertaking for two reasons. Firstly, a fundamental requirement of the DFT dictates that data samples must be contiguous, which for the case of two-dimensional images, corresponds to a rectangular region of pixel intensities.[1] Secondly, Fourier analysis regards spectral composition as a global phenomenon, and components detected via the transform are assumed to exist over the full extent of the image.

In the remainder of this section, we propose the use of a deconvolution algorithm to remove the artefacts introduced by the shape components, and thus provide a shape-invariant Fourier description of the texture content of non-rectangular image patches.

4.2 Background

Deconvolution is based on the following idea. A non-rectangular texture patch, $i(x,y)$, is assumed to be the product of applying a binary shape mask, $w(x,y)$, to an uncorrupted version of the texture, $t(x,y)$. By the convolution theorem, we can write:

$$i(x,y) = t(x,y) \times w(x,y) \stackrel{DFT}{\rightleftharpoons} T(u,v) \otimes W(u,v) = I(u,v) \tag{4}$$

If this convolution can in some way be undone, then we can get back to the Fourier representation of the uncorrupted version texture:

$$T(u,v) = I(u,v) \otimes^{-1} W(u,v), \tag{5}$$

where $A \otimes^{-1} B$ denotes the deconvolution of A with respect to B.

Deconvolution, however is ill-posed. The solution is not unique, and the problem becomes that of choosing a plausible value from the set of possible solutions. Deconvolution algorithms attempt to make an informed guess as to the value of unobservable samples on the basis of what information is available.

A number of deconvolution algorithms have been proposed. Maximum Entropy, Lucy-Richardson, and CLEAN are amongst the most commonly used techniques. Maximum entropy methods (e.g. [12]) attempt to select the solution that is the most consistent with the observable data, while simultaneously providing maximum entropy. Here, *entropy* is defined as an abstract concept which when maximised, produces a positive image with a compressed range of pixel values. This effectively introduces a smoothness constraint which favours the image that assumes the least about missing data. The

[1] In theory, it would be possible to adapt the Fourier transform to handle solid, convex regions. Partially filled and/or concave regions represent a far more challenging situation.

Lucy-Richardson deconvolution algorithm [7] employs Bayesian inference to maximise the likelihood of the reconstructed image. Successive estimates are applied to iteratively improve the appearance of the image.

Our approach is based on the CLEAN deconvolution algorithm, originally developed for aperture synthesis in radio astronomy [5]. Various refinements, initiated by [3], have since been proposed [10], and boast improved computational efficiency when dealing with large synthesis arrays. Others [9, 13] have tailored the algorithm towards the analysis of specialised forms of corrupted data. Our specific interest lies in the methodology surrounding CLEAN deconvolution. The algorithm focuses on restoring individual spectral elements and has since been shown to be equivalent to a least-squares fitting of sinusoids in the spatial domain [11]. This approach is particularly appealing for our work on texture analysis, as the reconstruction is based on the very elements we wish to examine.

4.3 Deconvolution Algorithm

It is assumed that the pure texture can be modelled by a number of harmonic components. The deconvolution proceeds by detecting which component, when corrupted by the shape mask, would provide the best explanation of the patterns observed in Fourier spectra. The component is then re-located to the *clean spectrum*, and its footprint erased from the corrupted spectrum. The process is repeated until only noise residuals are left in the corrupted spectrum.

We now formulate the algorithm for the case of 1-d signals, and later extend the result to accommodate 2-d texture images.

Consider the texture pattern, $t(x)$, comprising a single sinusoidal component, having amplitude A, frequency $\hat{u}$, and phase ϕ:

$$t(x) = A\cos(2\pi\hat{u}x + \phi). \tag{6}$$

The uncorrupted Fourier representation, $T(u)$, of this texture pattern is given by:

$$T(u) = a\delta(u-\hat{u}) + a^*\delta(u+\hat{u}), \tag{7}$$

where $a = \frac{A}{2}e^{i\phi}$. Let $i(x)$ be the observed form of $t(x)$, as viewed through the incomplete shape window $w(x)$. From Equation 4, we can write:

$$I(u) = T(u) \otimes W(u).$$

Expanding the convolution gives:

$$I(u) = aW(u-\hat{u}) + a^*W(u+\hat{u}) \tag{8}$$

The single component pair becomes distorted, forming additional peaks as dictated by the pattern of the spectral window, $W(u)$. To achieve the deconvolution of Equation 5, we need to determine the nature of the original harmonic component (i.e. $\hat{u}$ and a), given the corrupted Fourier spectrum, $I(u)$, and the shape mask, $W(u)$, that was responsible for the corruption.

The form of the shape mask guarantees that dominant peak in the spectral window, $W(u)$, will be the DC term ($u = 0$). In this case, while the peaks of the original harmonic component become distorted by the convolution, their actual position remains unaffected.

We can therefore determine the frequency of the original sinusoid, $\hat{u}$, by locating the peak component pair of $I(u)$. Before we can recover the amplitude and phase information, we need to consider how the spectral leakage caused by one peak contributes to the overall response observed at the other peak. Substituting for $u = \hat{u}$ in Equation 8 gives:

$$I(\hat{u}) = aW(0) + a^* W(2\hat{u}) \tag{9}$$

Using the conjugate symmetry, we rearrange to give:

$$a = \frac{I(\hat{u})W_0 - I^*(\hat{u})W(2\hat{u})}{W_0^2 - |W(2\hat{u})|^2} \tag{10}$$

Using this approach, we can recover the original harmonic component of $T(u)$, from the corrupted spectrum, $I(u)$, given that we know form of the corruption, $W(u)$. If the original signal comprises more than one harmonic component, then the process needs to be repeated. At each iteration, a single component is retrieved and re-located to the *clean spectrum*. The component's imprint is then removed leaving the residual spectrum. The process continues until either the level of the peak residual falls below a predetermined *noise* threshold, or the number of iterations has reached some reasonable limit.

When deconvolving multiple harmonic components, a further consideration is required. The extent of the peak detected in the corrupted spectrum is no longer purely the result of a single component pair. The distortion due to several components may contribute to the peak's response. Therefore, we assume that the detected component is only partially responsible for the observed peak, and introduce g, the clean gain, to specify the proportion of the component that is to be extracted. Typically the clean gain will lie in the range $0.1 < g < 1.0$. Low values will improve the stability of the algorithm, though at the expense of requiring extra iterations to fully extract the clean components.

We now summarise the algorithm, and re-formulate the equations for the case of 2-d data.

Let $R^i(u,v)$ be the ith residual, and $R^0(u,v) = I(u,v)$. Starting at $i = 1$:

1. - Find the frequency, $(\hat{u}^i, \hat{v}^i)$, of the ith clean component, from the maximum of $|R^{i-1}(u,v)|$
   - Calculate clean impulse, a^i, from:

$$a^i = g \cdot \frac{I(\hat{u}^i, \hat{v}^i)W_0 - I^*(\hat{u}^i, \hat{v}^i)W(2\hat{u}^i, 2\hat{v}^i)}{W_0^2 - |W(2\hat{u}^i, \hat{v}^i)|^2}$$

2. Generate the ith residual spectrum,

$$R^i(u,v) = R^{i-1} - (a^i W(u - \hat{u}^i, v - \hat{v}^i) + (a^i)^* W(u + \hat{u}^i, v + \hat{v}^i))$$

3. If the peak residual of $R^i(u,v)$ falls below the noise threshold, or if the maximum number of iterations has been reached, then proceed to step 4. If these termination conditions are not met, then continue cleaning from step 1

4. Construct the *clean spectrum*, C, from the K clean components:

$$C(u,v) = \sum_{i=1}^{K} a^i \delta(u - \hat{u}^i, v - \hat{v}^i) + (a^i)^* \delta(u + \hat{u}^i, v + \hat{v}^i)$$

The algorithm is demonstrated in Figure 3. Each of the non-rectangular texture patches has been deconvolved, and the result returned to the spatial domain for inspection. The left texture patch comprises a single harmonic component, and can be fully reconstructed by a single iteration (assuming the clean gain is set to 1.0), with zero error in the remaining residual. The other patches exhibit more realistic textures, comprising many spectral components. Although several iterations are required, the final reconstruction is impressive.

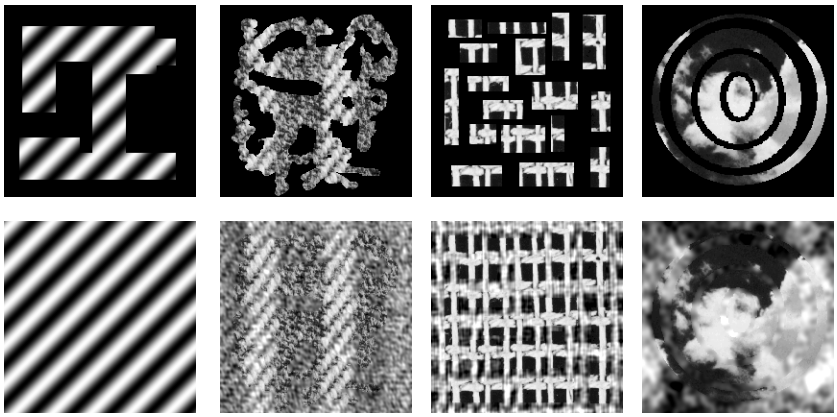

Figure 3: Texture Deconvolution. Arbitrarily-shaped texture patches (top), deconvolved to recover the *pure* texture image (bottom).

5 Deconvolution in Texture Analysis

5.1 Experiment

To assess the ability if the deconvolution algorithm to remove the detrimental effects of shape variation on the analysis of texture, the experiment described in section 3 was repeated. Deconvolution is applied to the Fourier transform prior to the projection of the magnitude components onto the texture features. The gain, g, and the noise threshold, t, were set to 0.5 and 1.0 respectively. The maximum number of iterations, I, was set to 500, although in practice this limit was rarely reached. In all other respects, the experiment was conducted as before.

5.2 Results and Discussion

The results of this experiment, together with the previous findings, are shown in Figure 4. It was found that the recognition system, applied to non-rectangular patches, performed significantly better when augmented with the deconvolution algorithm. Overall classification accuracy was improved by 40% to over 70%.

As predicted, the greatest improvements were found for patches exhibiting regular textures. It is perhaps fitting that the deconvolution algorithm is of most benefit for those

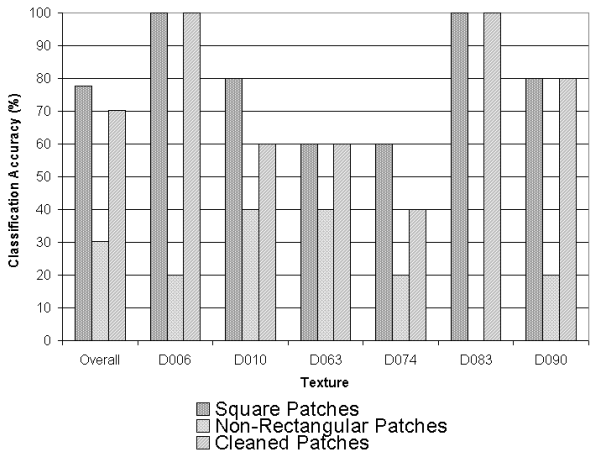

Figure 4: Texture recognition of deconvolved patches

textures that Fourier analysis is best equipped to deal with.

The results suggests that the deconvolution algorithm is capable of removing the artefacts introduced by shape components, and allows the extraction of a set of texture features that is largely invariant to shape variations. By including the deconvolution algorithm, the performance of the system is restored to a level approaching that attained for square patches.

We anticipate that similar benefits would be observed for other applications that make use of Fourier-based techniques for the analysis of image regions (e.g. region-based image coding, object recognition, extraction of texture maps from real image data, etc...).

6 Conclusions

Fourier analysis is a valuable tool for extracting texture information from image data, and a number of extensions have been proposed in the literature. However, due to the nature of the Fourier transform, many of these techniques are only applicable to the analysis of rectangular image patches. In situations where the texture patch does not entirely fill the region of analysis, information relating to the shape of the patch becomes entwined with its texture content thus contaminating the Fourier spectrum and corrupting the texture information. We have shown that Fourier-based texture features are indeed vulnerable to the artefacts introduced by shape components, responding to properties relating to the shape of the patch instead of its texture content. We have proposed the use of a deconvolution algorithm to remove these artefacts. The algorithm provides a shape-invariant Fourier description of the patch, permitting the extraction of robust texture features from non-rectangular image patches. In a texture recognition task involving the entire Brodatz set, classification performance was restored to a level similar to that attained for rectangular patches. We anticipate that similar benefits would be observed for other applications that make use of Fourier-based techniques.

References

[1] Alan Conrad Bovik, Marianna Clark, and Wilson S. Geisler. Multichannel texture analysis using localized spatial filters. *IEEE Trans. Pattern Analysis and Machine Intelligence*, PAMI-12(1):55–73, January 1990.

[2] Phil Brodatz. *Textures: A Photographic Album for Artists and Designers*. Dover, NY, 1966.

[3] B.G. Clark. An efficient implementation of the algorithm CLEAN. *Astronomy and Astrophysics*, 89:377–378, 1980.

[4] Robert M. Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804, May 1979.

[5] J. A. Hogbom. Aperture synthesis with a non-regular distribution of interferometer baselines. *Astronomy and Astrophysics, Supplement*, 15:417–426, 1974.

[6] Fang Liu and Rosalind W. Picard. Peridicity, directionality, and randomness: Wold features for image modeling and retrieval. *IEEE Trans. Pattern Analysis and Machine Intelligence*, PAMI-18(7):722–733, July 1996.

[7] L. B. Lucy. An iterative technique for the rectification of observed images. *The Astronomical Journal*, 79(6):745–754, 1974.

[8] Rosalind W. Picard and T. Kabir. Finding similar patterns in large image databases. In *Proc. Int. Conf. Acoust. Speech, Signal Proc.*, volume V, pages 161–164, Minneapolis, MN, April 1993.

[9] David H. Roberts, Joseph Lehár, and John W. Dreher. Time series analysis with CLEAN. I. derivation of a spectrum. *The Astronomical Journal*, 93(4):968–989, 1987.

[10] F. R. Schwab. Relaxing the isoplanatism assumption in self-calibration: Applications to low-frequency radio interferometry. *The Astronomical Journal*, 89:1076–1081, 1984.

[11] U. J. Schwarz. Mathematical-statistical description of the iterative beam removing technique Method CLEAN. *Astronomy and Astrophysics*, 65:345–356, 1978.

[12] J. Skilling and R. K. Bryan. Maximum entropy image reconstruction: General algorithm. *Monthly Notices of the Royal Astronomical Society*, 211:111–124, 1984.

[13] D. G. Steer, P. E. Dewdney, and M. R. Ito. Enhancements to the decovolution algorithm CLEAN. *Astronomy and Astrophysics*, 137:159–165, 1984.

[14] Hideyuki Tamura, Shunji Mori, and Takashi Yamawaki. Textural features corresponding to visual perception. *IEEE Trans. Systems, Man, and Cybernetics*, 8(6):460–473, 1978.