TABLE I
VAX 11/780 CPU TIME PER RUN IN ms

Number of Point Correspondences	Method Used		
	SVD	Quaternion	Iterative
3	54.6	26.6	126.8 (25)
7	41.6	32.4	108.2 (12)
11	37.0	41.0	105.2 (8)
16	39.4	45.6	94.2 (5)
20	40.4	45.2	135.0 (6)
30	44.2	48.3	111.0 (6)

used in finding the SVD (subroutine LSVDF) and in doing the eigen analysis (subroutine EIGRS) for the quaternion method. For the iterative method, the initial guess solution was zero in all cases.

We observe that the computer time requirements of the SVD and the quaternion algorithms are comparable, while the time for the iterative method is much longer. However, in the iterative method, the solutions were calculated to 7-digit accuracy. If we can accept 10 percent accuracy, then the number of iterations are reduced by a factor of 2 to 3. Furthermore, the rate of convergence can be increased by overrelaxation.

REFERENCES

[1] S. D. Blostein and T. S. Huang, "Estimating 3-D motion from range data," in *Proc. 1st Conf. Artificial Intelligence Applications*, Denver, CO, Dec. 1984, pp. 246–250.
[2] D. Cyganski and J. A. Orr, "Applications of tensor theory to object recognition and orientation determination," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-7, pp. 663–673, Nov. 1985.
[3] T. S. Huang, S. D. Blostein, and E. A. Margerum, "Least-squares estimation of motion parameters from 3-D point correspondences," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Miami Beach, FL, June 24–26, 1986.
[4] O. D. Faugeras and M. Hebert, "A 3-D recognition and positioning algorithm using geometrical matching between primitive surfaces," in *Proc. Int. Joint Conf. Artificial Intelligence*, Karlshrue, West Germany, Aug. 1983, pp. 996–1002.
[5] M. Fischler and Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, June 1981.

Registration of Translated and Rotated Images Using Finite Fourier Transforms

E. DE CASTRO AND C. MORANDI

***Abstract*—A well-known method for image registration is based on a conventional correlation between phase-only, or whitened, versions of the two images to be realigned. The method, covering rigid translational movements, is characterized by an outstanding robustness against correlated noise and disturbances, such as those encountered with nonuniform, time varying illumination. This correspondence discusses an extension of the method to cover both translational and rotational movements.**

***Index Terms*—Digital image processing, fast Fourier transform, image registration, image sequence analysis, motion estimation.**

Manuscript received October 11, 1984. Recommended for acceptance by S. W. Zucker.

E. De Castro was with the Dipartimento di Elettronica, Informatica e Sistemistica, Università di Bologna, Viale Risorgimento 2, 40136 Bologna, Italy.

C. Morandi is with the Dipartimento di Elettronica ed Automatica, Università di Ancona, via Brecce Bianche, 60100 Ancona, Italy.

IEEE Log Number 8715507.

I. INTRODUCTION

Let us consider a plane image performing rigid movements of translation and rotation within a rectangular domain C representing the observed field. Let the image be defined by a density function vanishing outside a region A_t whose position varies with time t, but is always fully contained inside C. The limitations which arise when parts of the image leave the observed field will be mentioned later on.

Let $s_0(x, y)$ represent the image at a reference time, $t = 0$, and $s_t(x, y)$ be the present image, which is but a replica of $s_0(x, y)$ translated by (x_0, y_0) and rotated by θ_0:

$$s_t(x, y) = s_0(u, v), \tag{1}$$

where

$$x - x_0 = u \cos\theta_0 - v \sin\theta_0$$
$$y - y_0 = u \sin\theta_0 + v \cos\theta_0. \tag{2}$$

In order to realign images s_0 and s_t, it is first necessary to determine the translation vector (x_0, y_0) and the rotation θ_0 from the information provided by $s_0(x, y)$ and $s_t(x, y)$. Several image registration algorithms are known: for excellent review papers see [1]–[6].

The present correspondence is meant as a contribution in the context of the phase correlation technique [7], [8], which was developed with reference to purely translatory displacements. According to [7], [8], let $S_0(\xi, \eta)$ and $S_t(\xi, \eta)$ be the Fourier transforms of $s_0(x, y)$ and $s_t(x, y)$. Since in the case of pure translations $s_t(x, y) = s_0(x - x_0, y - y_0)$, it follows that

$$S_t(\xi, \eta) = e^{-j2\pi(\xi x_0 + \eta y_0)} S_0(\xi, \eta). \tag{3}$$

Therefore, by inverse transforming the ratio of the cross-power spectrum of s_t and s_0 to its magnitude, $S_t S_0^* / |S_t S_0^*| = \exp(-j2\pi(\xi x_0 + \eta y_0))$, a Dirac δ-distribution centered on (x_0, y_0) is obtained. In practice continuous transforms are replaced by finite ones, and by inverse transformation a unity pulse centered on (x_0, y_0) is obtained, so that the translation is immediately determined.

In this correspondence the principles of a generalization of the phase correlation method for the registration of rotated and translated images [9] are briefly recalled and the corresponding numerical algorithm is presented. The effectiveness of the procedure is then illustrated by means of simple experiments. Finally, the edge effects which arise when the image completely fills the observed field C are pointed out.

This correspondence is a part of research aiming at the implementation of an image stabilization system [10] for the observation of the human retina, in which the image viewed by the TV camera may be observed on a monitor free of the unavoidable spontaneous movements, which do not allow any automatic analysis of dynamic effects, such as the pulsations frequently observed in blood vessels. The program is carried out in cooperation with the Ophthalmological Clinic of the University of Bologna and the IBM Research Center in Rome.

II. THEORY

If $s_t(x, y)$ is a translated and rotated replica of $s_0(x, y)$, [see (1)], according to the Fourier Shift Theorem and the Fourier Rotation Theorem [11] their transforms are related by

$$S_t(\xi, \eta) = e^{-j2\pi(\xi x_0 + \eta y_0)} S_0(\xi \cos\theta_0 + \eta \sin\theta_0, -\xi \sin\theta_0 + \eta \cos\theta_0) \tag{4}$$

Let us then consider the ratio

$$G(\xi, \eta; \theta) = \frac{S_t(\xi, \eta)}{S_0(\xi \cos\theta + \eta \sin\theta, -\xi \sin\theta + \eta \cos\theta)}, \quad (5)$$

with θ taken as a variable. Obviously, for $\theta = \theta_0$

$$G(\xi, \eta; \theta_0) = e^{-j2\pi(\xi x_0 + \eta y_0)} \quad (6)$$

while, as the difference between θ and θ_0 increases, $G(\xi, \eta; \theta)$ will increasingly differ from this exponential form.

The intuitive meaning of (6) is easily understood: in fact, the rotated version of S_0, $S_0(\xi \cos\theta + \eta \sin\theta, -\xi \sin\theta + \eta \cos\theta)$ is the transform of the image obtained by rotating s_0 by θ [11]. Consequently, if $\theta = \theta_0$ the problem is reduced to the tracking of a purely translational movement.

The procedure consists therefore of determining first the angle $\theta = \theta_0$ for which $G(\xi, \eta; \theta)$ reduces to the form (6), then in evaluating x_0 and y_0 as in the case of pure translations.

The 2-D transforms required by the above procedure may, in principle, be reduced to 1-D following the method outlined in [12]. For instance, letting $\eta = 0$ in (5) and (6), we find

$$G(\xi, 0; \theta) = \frac{S_t(\xi, 0)}{S_0(\xi \cos\theta, -\xi \sin\theta)}$$

$$G(\xi, 0; \theta_0) = e^{-j2\pi\xi x_0} \quad (7)$$

and θ_0 and x_0 are easily determined. In a quite similar way θ_0 and y_0 are determined. However, the reduced robustness of the one-dimensional method, combined with some resampling problems to be mentioned later on, renders the 1-D approach less attractive for rotated images.

Notice that, in the above derivation, the ratio S_t/S_0 was considered for simplicity of notation, instead of the cross-power spectrum divided by its magnitude used in [7]. It is immediately seen that the latter is the cross-power spectrum of the phase-only images obtained from s_0 and s_t by forcing to unity the magnitude of their Fourier transforms. Although it is possible, in theory, to devise situations where the use of S_t/S_0 may be of advantage, and surely there is no detriment if both A_0 and A_t are fully contained in C and noise is absent, in practice it is always convenient to follow the original approach, for a number of reasons clearly outlined in [7], [8]. However, for ease of notation, we shall proceed using S_t/S_0.

III. Numerical Algorithm

Evaluation of $G(\xi, \eta; \theta)$ requires a numerical evaluation of the transformation integrals yielding $S_0(\xi, \eta)$ and $S_t(\xi, \eta)$, which may be limited to the observation field C under our assumptions. To this end, we shall define in C a rectangular mesh with nodes in $(m\Delta x, n\Delta y)$, $m = 0, 1, \cdots, M-1$, $n = 0, 1, \cdots, N-1$. Obviously, if X and Y are the x- and y-dimensions of C, $M\Delta x = X$ and $N\Delta y = Y$.

The Fourier transform of a function $r(x, y)$ vanishing outside C may be approximated by

$$R(\xi, \eta) = \int\int_C r(x, y)\, e^{-j2\pi(\xi x + \eta y)}\, dx\, dy \approx \Delta x \Delta y \tilde{R}(\xi, \eta),$$

$$\tilde{R}(\xi, \eta) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} r(m\Delta x, n\Delta y)\, e^{-j2\pi(\xi m\Delta x + \eta n\Delta y)}. \quad (8)$$

If a discretization is also performed in the frequency domain by a rectangular mesh with nodes in $(k/(M\Delta x), l/(N\Delta y))$, $k = 0, 1, \cdots, M-1$; $l = 0, 1, \cdots, N-1$, it turns out that:

$$\tilde{R}\big(k/(M\Delta x), l/(N\Delta y)\big) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} r(m\Delta x, n\Delta y) \cdot \exp\big(-j2\pi(mk/M + nl/N)\big) = R_d(k, l) \quad (9)$$

where $R_d(k, l)$ is the finite Fourier transform of the finite sequence

$$r_d(m, n) = r(m\Delta x, n\Delta y). \quad (10)$$

Let us now consider, in the frequency domain, the two points (ξ, η) and (μ, ν) where $\mu = \xi \cos\theta + \eta \sin\theta$ and $\nu = -\xi \sin\theta + \eta \cos\theta$. According to (8) it can be assumed that

$$G(\xi, \eta; \theta) = \frac{S_t(\xi, \eta)}{S_0(\mu, \nu)} \approx \frac{\tilde{S}_t(\xi, \eta)}{\tilde{S}_0(\mu, \nu)}. \quad (11)$$

To turn to finite transforms, let point (ξ, η) coincide with node (k, l) of the mesh in the frequency plane. Point (μ, ν) will not, in general, coincide with any node of the same mesh, hence it will be necessary to replace $S_0(\mu, \nu)$ by some suitable interpolated value. Here, the value of S_0 in the nearest node, $(p(k, l), q(k, l))$ was used, the integers p and q obviously being functions of k, l for any given θ. It is now straightforward to rewrite (11) as

$$G_d(k, l; \theta) = \frac{S_{td}(k, l)}{S_{0d}\big(p(k, l), q(k, l)\big)} \quad (12)$$

which may always be inverse-transformed. If $\theta = \theta_0$ then

$$G_d(k, l, \theta_0) \approx e^{-j2\pi(m_0 k/M + n_0 l/N)} \quad (13)$$

where (m_0, n_0) is a node of the mesh near (x_0, y_0), and it is immediately recognized that its inverse transform $g_d(m, n; \theta_0)$ is a unity pulse centered on (m_0, n_0).

θ_0 may thus be determined by varying θ until the shape of $g_d(m, n; \theta)$ gives the closest approximation of a unity pulse. Of course, possible symmetries of the image may render θ_0 not uniquely defined, and even in the absence of symmetries ambiguities may arise when a rotation cannot be distinguished from a suitable translation. In general, however, it is easy to recognize such situations.

The algorithm is now fully defined: some caution however is required in determining the maximum acceptable values of Δx and Δy, which should meet the sampling theorem requirements. It seems that the search for θ_0, x_0, and y_0 may be performed in a fully automatic way, eventually using a directed search strategy for the determination of θ.

It should be stressed that in the above algorithm the tentative values of θ are introduced in the frequency domain through μ, ν in (11), i.e., when the transforms of $s_0(x, y)$ and $s_t(x, y)$ are already computed. One could, alternatively, rotate the image $s_0(x, y)$ by introducing the tentative values of θ through u, v before transforming, but this would imply transforming each rotated replica of $s_0(x, y)$, clearly at the expense of increased computational cost.

IV. Experimental Results and Concluding Remarks

Before discussing the experimental results, we would first recall that in the expression (12) of $G_d(k, l; \theta)$ μ and ν are replaced by the frequencies corresponding to the nearest node $(p(k, l), q(k, l))$ of the mesh on which S_0 is evaluated (nearest neighbor approximation). During the experiments it appeared necessary to use a better approximation of $S_0(\mu, \nu)$, and to this aim S_0 was evaluated on a 512×512 mesh, four times denser than the one used for S_t. This was obtained by the well-known technique of zero-padding the sampled data matrix which describes s_0. In fact, it is very important to find a satisfactory interpolation scheme: with a 128×128 mesh, for instance, neither nearest neighbor nor bilinear interpolation schemes were successful.

The proposed method was successfully applied in a series of experiments with synthetic and real images of various types. As a first example, we report the results obtained with a 128×128 image made of stripes, obtained by the superposition of shifted Gaussian functions with circular symmetry (see Fig. 1).

Typical results of the experiments are reported in the 3-D plots of Fig. 2, which represent $g_d(m, n; \theta)$ for several values of θ close to $\theta_0 = 10°$. It is apparent that the method is rather sensitive, as is demonstrated by the simultaneous increase of the pulse amplitude and decrease of the average disturbance power. It is interesting,

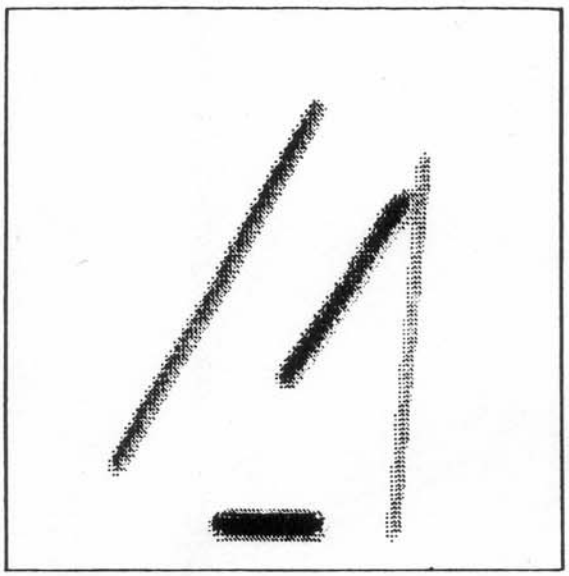

Fig. 1. The test image used for the experiments, in which it was assumed $x_0 = 2$ pixels, $y_0 = 5$ pixels, $\theta_0 = 10°$.

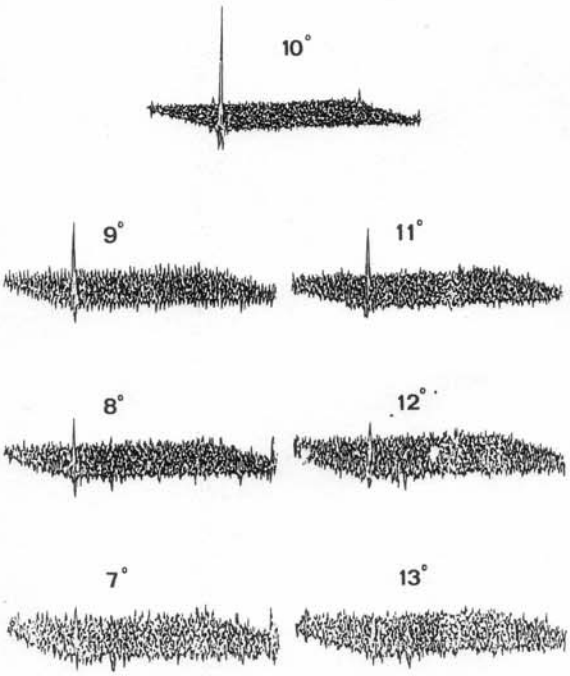

Fig. 2. The aspect of the corresponding $g_d(m, n; \theta)$ for various values of θ between 7° and 13°.

however, to consider the dependence of the main peak amplitude on θ in a sufficiently narrow interval centered on θ_0. As is shown in Fig. 3, the amplitude variation is rather regular: this suggests that, once the above-mentioned interval is found, directed search strategies may be used for the determination of θ_0. The interval itself cannot but be determined by nonoriented search techniques, for instance by successive halvings, unless the specific application suggests a sufficiently approximate first tentative value. In the envisioned application to ophthalmology, we expect rotations not exceeding $\pm 2°$ around the rest position.

Other experiments were performed on images of pebbles obtained by shifting and rotating the TV camera. Fig. 4 shows a typical reference image, while Fig. 5 demonstrates the increase in amplitude of the registration peak as the correct rotation angle, 19°, is approached. In this experiment a (22, 57) pixel displacement was measured using 128 × 128 images.

Finally, it should be pointed out that the assumption that A_t is

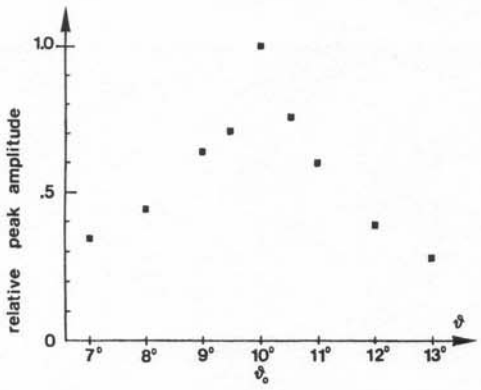

Fig. 3. Amplitudes of the peaks in Fig. 2 as a function of θ.

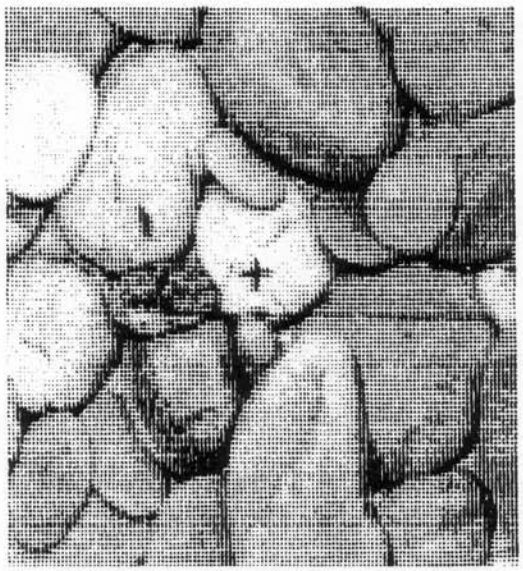

Fig. 4. The 128 × 128 pebbles image.

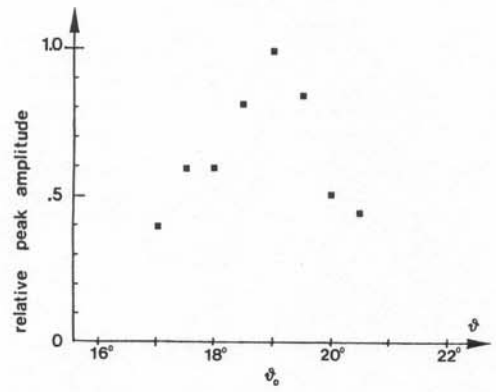

Fig. 5. Amplitude of the registration peak as a function of θ.

fully contained in C does not hold in the specific application which originated this work, as well as in the second of the examples shown. In fact, the whole observation field of the TV camera is filled by the image. So, when a part of the image leaves C through an arc of its contour, an identical part does not enter C through the opposite arc, and this causes an error. However, in a set of experiments performed on digitized retina images [13], in which only the translational component of the movement was compensated using the original phase correlation algorithm, it was observed that this error may be reduced to an acceptable level by simply windowing the image before the application of the algorithm, using a window with rotational symmetry.

References

[1] J. K. Aggarwal, L. S. Davis, and W. N. Martin, "Correspondence processes in dynamic scene analysis," *Proc. IEEE*, vol. 69, pp. 562–572, 1981.

[2] T. S. Huang and R. Y. Tsai, in *Image Sequence Analysis*, T. S. Huang, Ed. Berlin: Springer-Verlag, 1981, pp. 1–18, ch. 1.

[3] H. H. Nagel, "Image sequence analysis: What can be learnt from applications," in *Image Sequence Analysis*, T. S. Huang, Ed. Berlin: Springer-Verlag, 1981, pp. 19–228.

[4] ——, "Overview on image sequence analysis," in *Image Sequence Processing and Dynamic Scene Analysis*. Berlin: Springer-Verlag, 1983, pp. 2–39.

[5] J. K. Aggarwal and W. N. Martin, "Dynamic scene analysis," in *Image Sequence Processing and Dynamic Scene Analysis*. Berlin: Springer-Verlag, 1983, pp. 40–73.

[6] C. Cafforio and F. Rocca, "The differential method for image motion estimation," in *Image Sequence Processing and Dynamic Scene Analysis*. Berlin: Springer-Verlag, 1983, pp. 104–124.

[7] C. D. Kuglin and D. C. Hines, "The phase correlation image alignment method," in *Proc. 1975 Int. Conf. Cybernetics and Society*, Sept. 1975, pp. 163–165.

[8] J. J. Pearson, D. C. Hines, S. Golosman, and C. D. Kuglin, "Video rate image correlation processor," *Proc. SPIE, (Application of Digital Image Processing, IOCC 1977)*, vol. 119, pp. 197–205.

[9] E. De Castro and C. Morandi, "Tracking di immagini in movimento rototraslatorio mediante trasformate di Fourier," *Trans. Accademia delle Scienze dell' Istituto di Bologna*, Apr. 1984.

[10] E. De Castro, G. Cristini, A. Giovannini, C. Morandi, and E. Sangiorgi, "Sopra alcuni problemi connessi all' esplorazione televisiva del fondo oculare," *Atti della Accademia delle Scienze dell' Istituto di Bologna*, Rendiconti, serie XIII, tomo IX, pp. 233–244, 1982.

[11] A. Papoulis, *Systems and Transforms with Application in Optics*. New York: McGraw-Hill, 1968.

[12] S. Alliney and C. Morandi, "Digital image registration using projections," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, pp. 222–233, 1986.

[13] E. De Castro, C. Morandi, G. Cristini, A. Martelli, and M. Vascotto, "Feasibility of an electronic ophthalmoscope with compensation of random eye motion," in *Advances in Image Processing and Pattern Recognition*, V. Capellini and R. Marconi, Eds. Amsterdam, The Netherlands: North-Holland, 1986, pp. 328–333.

Multispectral Texture Synthesis Using Fractal Concepts

NIGEL DODD

***Abstract*—Natural textures are synthesized across several spectral bands using no more than a single second-order descriptor for each band. The Fractal model for monochrome textures is used as a null-hypothesis. The eye's sensitivity to Fractal dimension under specific conditions is determined. Multispectral camouflage is generated to imitate a range of natural textures.**

***Index Terms*—Camouflage, eigencomponent, Fractals, multispectral, texture.**

I. Monochrome Fractal Textures

Natural form is often the result of cumulative actions of the same type. When the magnitude of the actions is rigorously ordered according to scale, the result is describable by the Fractal model [1]. Under these circumstances the second-order statistics are described by a single number: the Fractal dimension. For the efficiency of specification and synthesis of natural texture there is much to recommend the Fractal concept. The purpose of this research has been to determine how closely nature adheres to the Fractal model when painting her foliage patterns, and how convincing imitations based solely on this descriptor appear to be.

A. Specification

For scalar brightness variations as a function of two dimensions, the texture is Fractal if

$$\langle |\Delta I| \rangle \, \|\Delta x\|^{-H} = \text{constant} \qquad (1)$$

where $\langle |\Delta I| \rangle$ is the expected magnitude of brightness variation over a displacement Δx. H characterizes the second-order statistics.

B. Generation

In order to perform visual comparisons between natural and synthetic textures a number of methods were employed to synthesize Fractal, or near-Fractal, textures. The details and shortcomings of the methods summarized below are well documented [1], [2].

1) Fourier Transform Method: A random Gaussian field of scalar brightness values is spatially filtered to generate well defined second-order connections between pixels. The phase of the Fourier transformed random field is unaltered, but the modulus is forced to assume a form f^{-P}. The parameter P can be altered to cause the inverse Fourier transform of this modified field to have any particular Fractal dimension between 2 and 3.

2) Cylindrical Integration Method: A path within a random Gaussian field of zero mean painted on a cylinder is integrated over to produce a Brownian variation in image brightness. The path, in cylindrical coordinates, is defined as $(x \cos(\theta) + y \sin(\theta), \theta)$, where (x, y) is the rectangular coordinate of the cell in the Brownian field, and θ is the variable of integration. Although able to produce true Fractals to the precision of the implementation, this method is unable to generate a range of different dimension Fractals. It is useful, however, as a reference image generator for analysis algorithms.

3) Midpoint Displacement Method: The brightness at the center of a square is interpolated from the four corner brightness values. This value is perturbed by an amount related to the size of the square. The four subsquares are treated similarly and so on down to the resolution of the frame-store. This is detailed in [4]. In the reply to [4], and also in [2], Mandelbrot [5] shows that the result is not truly Fractal. However this method is quick, requiring of order N calculations (N being the number of pixels), which, apart from the random perturbation, require only addition and logical word shifting. Also, by defining the perturbation modulus individually at each stage, it is possible to represent better some textures which do not accurately fit the Fractal definition and which contain thresholds of scale beyond which certain features become apparent.

C. Analysis of Dimension

Many algorithms exist for determining the Fractal dimension of curves drawn in two-space, e.g., coastlines. Some of these have their analogs in space-to-line functions, e.g., geographical relief, brightness textures. One of these, implemented by Peleg *et al.* [6], was used to measure the Fractal dimension of natural and artificial texture. Additionally, an algorithm bearing a closer tie with conventional cooccurrence methods, and described by Pentland [7], was used for comparison.

1) Peleg: If the brightness field of a texture is imagined to form a surface with brightness being represented by height, then the method of [6] is based on the volume v, occupied by all points distant by ϵ, or less, from the surface. The surface area is obtained

Manuscript received April 18, 1986; revised November 10, 1986.

The author is with the Research Initiative in Pattern Recognition, RSRE, St. Andrews Road, Malvern WR14 3PS, England.

IEEE Log Number 8715804.