Two-Dimensional Fourier Series-Based Model for Nonminimum-Phase Linear Shift-Invariant Systems and Texture Image Classification

Chii-Horng Chen, Chong-Yung Chi, *Senior Member, IEEE*, and Ching-Yung Chen

***Abstract*—In this paper, Chi's real one-dimensional (1-D) parametric nonminimum-phase Fourier series-based model (FSBM) is extended to two-dimensional (2-D) FSBM for a 2-D nonminimum-phase linear shift-invariant system by using finite 2-D Fourier series approximations to its amplitude response and phase response, respectively. The proposed 2-D FSBM is guaranteed stable, and its complex cepstrum can be obtained from its amplitude and phase parameters through a closed-form formula without involving complicated 2-D phase unwrapping and polynomial rooting.**

A consistent estimator is proposed for the amplitude estimation of the 2-D FSBM using a 2-D half plane causal minimum-phase linear prediction error filter (modeled by a 2-D minimum-phase FSBM), and then, two consistent estimators are proposed for the phase estimation of the 2-D FSBM using the Chien *et al.* 2-D phase equalizer (modeled by a 2-D allpass FSBM). The estimated 2-D FSBM can be applied to modeling of 2-D non-Gaussian random signals and 2-D signal classification using complex cepstra. Some simulation results are presented to support the efficacy of the three proposed estimators. Furthermore, classification of texture images (2-D non-Gaussian signals) using the estimated FSBM, second-, and higher order statistics is presented together with some experimental results. Finally, we draw some conclusions.

***Index Terms*—Higher order statistics, 2-D Fourier series-based model, 2-D non-Gaussian signals, 2-D nonminimum-phase linear shift-invariant systems.**

I. INTRODUCTION

TWO-DIMENSIONAL (2-D) parametric models for linear shift-invariant (LSI) systems such as autoregressive (AR), moving average (MA) and autoregressive moving average (ARMA) models, have been widely used in a variety of 2-D statistical signal processing applications, such as 2-D spectral estimation, texture image synthesis, and classification [1]–[4], [6]–[9], [11]. Usually, model parameters are estimated from the given 2-D data (e.g., digital images) using a stationary random field model, and then, the estimated model parameters are further used in the application of interest. There have been a number of algorithms reported for the estimation of model parameters. In [1]–[3], 2-D AR parameters are estimated using the least-squares (LS) estimator based on a set of linear equations formed from autocorrelations [second-order statistics (SOS)] of the given 2-D data. Assuming that the given 2-D data are Gaussian, Kashyap and Chellappa [2] proposed an approximate maximum-likelihood (AML) algorithm that iteratively updates 2-D AR parameter estimates by solving a set of linear equations also formed from autocorrelations of the given 2-D data to avoid excessive computational load. In [4], the 2-D DFT of 2-D MA model is obtained by a linear geometric transform of a one-dimensional (1-D) function. With the assumption that the 2-D DFT of the given 2-D data is independent identically distributed (i.i.d.) complex Gaussian, the 1-D function is estimated using an ML algorithm. Then, the 2-D MA model obtained from the estimated 1-D function is applied to synthesis of texture images in [4]. Due to the fact that SOS are blind to the system phase, these approaches cannot completely characterize 2-D data, and thus, their performance can be limited without using the system phase information.

Higher order statistics (HOS) known as cumulants [5], which include both amplitude and phase information of non-Gaussian random fields, have been used for the estimation of 2-D nonminimum-phase asymmetric noncausal AR or ARMA models [6]–[8], [11]. For instance, as reported in [8], Gaussianity and linearity tests indicate that a texture image can be modeled as a 2-D LSI system (2-D texture image model) driven by an i.i.d. non-Gaussian random field. Inverse filter criteria have been proposed for estimating parameters of ARMA models [6], [7]. Tugnait [6] proposed three inverse filter criteria for jointly estimating AR and MA parameters, whereas only AR parameters are used for texture synthesis. Hall and Giannakis [7] proposed two inverse filter criteria for estimating AR parameters, whereas MA parameters are estimated either by a closed-form solution using cumulants or by cumulant matching. Then, estimated AR parameters are used as a feature vector for texture image classification. Hall and Giannakis [8] also estimate ARMA parameters by polyspectral matching, whereas only AR parameters are used for texture image synthesis. Moreover, Tsatsanis and Giannakis [9] also proposed a nonparametric cumulant matching method for texture image classification. Recently, Chi and Chen [10] proposed a nonparametric 2-D frequency domain blind system identification algorithm with application to texture synthesis. These methods can characterize a broader class of texture image representations than phase-insensitive SOS-based approaches, whereas AR parameters appear more suitable for texture image classification and synthesis than both the MA and ARMA models.

Manuscript received August 19, 1999; revised December 6, 2001. This work was supported by the National Science Council of taiwan, R.O.C., under Grant NSC 87-2213-E007-026. Part of this work was presented at the IEEE Signal Processing Workshop on Higher Order Statistics, Ceasarea, Israel, June 14–16, 1999. The associate editor coordinating the review of this paper and approving it for publication was Prof Jian Li.

The authors are with the Department of Electrical Engineering and Institute of Communications Engineering, National Tsing Hua University, Hsinchu, Taiwan, R.O.C. (e-mail: cychi@ee.nthu.edu.tw).

Publisher Item Identifier S 1053-587X(02)02380-2.

Recently, Chi [12], [13] proposed a real 1-D Fourier series-based model (FSBM) as an approximation to an arbitrary nonminimum-phase linear time-invariant (LTI) system. Chi's 1-D FSBM is potentially preferable to the 1-D ARMA model in 1-D statistical signal processing applications because of the following two characteristics of the former.

C1) The 1-D FSBM, which can be causal or noncausal, minimum-phase or nonminimum-phase, is guaranteed stable.

C2) Without involving phase unwrapping and polynomial rooting (needed for finding poles and zeros of the 1-D ARMA model), the complex cepstrum of the 1-D FSBM can be easily obtained from the 1-D FSBM parameters via a closed-form formula.

In this paper, a real 2-D FSBM, which is a straightforward extension of the 1-D FSBM, is proposed for approximation to an arbitrary 2-D nonminimum-phase LSI system. Characteristics C1) and C2) of the 1-D FSBM also apply to the 2-D FSBM. Then, we present identification and estimation of the 2-D FSBM with the given 2-D non-Gaussian data followed by its application to classification of texture images.

The paper is organized as follows. Section II presents the real 2-D FSBM. Then, three iterative algorithms for estimating the parameters of the 2-D FSBM are presented in Section III, together with the consistency of the three estimators. In Section IV, some simulation results are presented to support the efficacy of the three proposed algorithms. Section V presents texture image classification using the estimated 2-D FSBM parameters, second-, and higher order statistics, followed by some experimental results with real texture image data. Finally, we draw some conclusions.

II. Two-Dimensional FSBM

Assume that $h(m,n)$ is a real stable 2-D LSI system with the frequency response $H(\omega_1,\omega_2)$. The proposed 2-D FSBM for $H(\omega_1,\omega_2)$ can be expressed as the following two decompositions.

A. Two-Dimensional Magnitude (MG)-Phase (PS) Decomposition

$$\begin{aligned} H(\omega_1,\omega_2) &= H^*(-\omega_1,-\omega_2) \\ &= H_{\rm MG}(\omega_1,\omega_2)\cdot H_{\rm PS}(\omega_1,\omega_2) \end{aligned} \tag{1}$$

where $H_{\rm MG}(\omega_1,\omega_2)$ is a 2-D zero-phase FSBM given by

$$\begin{aligned} H_{\rm MG}&(\omega_1,\omega_2) \\ &= H_{\rm MG}(-\omega_1,-\omega_2) \\ &= \exp\left\{\sum\sum_{(i_1,i_2)\in\Omega(p_1,p_2)} \alpha_{i_1,i_2}\cos(i_1\omega_1+i_2\omega_2)\right\} \end{aligned} \tag{2}$$

and $H_{\rm PS}(\omega_1,\omega_2)$ is a 2-D allpass FSBM given by

$$\begin{aligned} H_{\rm PS}&(\omega_1,\omega_2) \\ &= \frac{1}{H_{\rm PS}(-\omega_1,-\omega_2)} \\ &= \exp\left\{j\sum\sum_{(i_1,i_2)\in\Omega(p_1,p_2)} \beta_{i_1,i_2}\sin(i_1\omega_1+i_2\omega_2)\right\} \end{aligned} \tag{3}$$

where $\Omega(p_1,p_2)$, which is the region of support for both the real amplitude parameters α_{i_1,i_2} and real phase parameters β_{i_1,i_2}, is a truncated nonsymmetric half plane (TNSHP) [14], [15] given by

$$\begin{aligned} \Omega(p_1,p_2) = \{(i_1,i_2) &: i_1=1,\ldots,p_1, i_2=-p_2,\ldots,p_2\} \\ &\cup\{(i_1,i_2): i_1=0, i_2=1,\ldots,p_2\} \end{aligned} \tag{4}$$

to which (0, 0) does not belong.

B. Two-Dimensional Minimum-Phase (MP)-Allpass (AP) Decomposition

The 2-D FSBM given by (1) can also be expressed as

$$\begin{aligned} H(\omega_1,\omega_2) &= H^*(-\omega_1,-\omega_2) \\ &= H_{\rm MP}(\omega_1,\omega_2)\cdot H_{\rm AP}(\omega_1,\omega_2) \end{aligned} \tag{5}$$

where $H_{\rm MP}(\omega_1,\omega_2)$ is a 2-D minimum-phase FSBM given by

$$\begin{aligned} H_{\rm MP}(\omega_1,\omega_2) &= H^*_{\rm MP}(-\omega_1,-\omega_2) \\ &= \exp\left\{\sum\sum_{(i_1,i_2)\in\Omega(p_1,p_2)} \alpha_{i_1,i_2}e^{-j(i_1\omega_1+i_2\omega_2)}\right\} \end{aligned} \tag{6}$$

and $H_{\rm AP}(\omega_1,\omega_2)$ is a 2-D allpass FSBM given by

$$\begin{aligned} H_{\rm AP}(\omega_1,\omega_2) &= \frac{1}{H_{\rm AP}(-\omega_1,-\omega_2)} \\ &= \exp\left\{j\sum\sum_{(i_1,i_2)\in\Omega(p_1,p_2)} (\alpha_{i_1,i_2}+\beta_{i_1,i_2}) \right. \\ &\qquad \left. \times\ \sin(i_1\omega_1+i_2\omega_2)\right\}. \end{aligned} \tag{7}$$

It can be easily shown that the region of support of the minimum-phase system $h_{\rm MP}(m,n)$ given by (6) is the right half plane (i.e., $\Omega(\infty,\infty)$) and $h_{\rm MP}(0,0)=1$.

The 2-D FSBM given by (1) and (5) is potentially a better choice for modeling arbitrary 2-D LSI systems than the 2-D ARMA model in statistical signal processing applications mentioned in Section I due to two characteristics that are discussed as follows.

C3) Because $H(\omega_1,\omega_2)$ (with parameters α_{i_1,i_2} and β_{i_1,i_2}) is a 2-D continuous periodic function of ω_1 and ω_2 with the same period of 2π, the LSI system $h(m,n)$ is absolutely summable by the property of Fourier series and, thus, is stable. Moreover, the inverse system $1/H(\omega_1,\omega_2)$ is also a 2-D FSBM (with parameters $-\alpha_{i_1,i_2}$ and $-\beta_{i_1,i_2}$) and, thus, is stable as well.

C4) Let $\tilde{h}(m,n)$ denote the complex cepstrum of $h(m,n)$, i.e., the 2-D inverse Fourier transform of $\ln\{H(\omega_1,\omega_2)\}$ [16]–[18]. It can be easily shown from the MP-AP decomposition of the 2-D FSBM given by (5) that

$$\tilde{h}(m,n) = \tilde{h}_{\rm MP}(m,n) + \tilde{h}_{\rm AP}(m,n) \tag{8}$$

where

$$\tilde{h}_{\rm MP}(m,n) = \begin{cases} \alpha_{m,n}, & (m,n)\in\Omega(p_1,p_2) \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

and

$$\tilde{h}_{\rm AP}(m,n) = \begin{cases} -(1/2)(\alpha_{m,n}+\beta_{m,n}), & (m,n)\in\Omega(p_1,p_2) \\ (1/2)(\alpha_{-m,-n}+\beta_{-m,-n}), & (-m,-n)\in\Omega(p_1,p_2) \\ 0, & \text{otherwise.} \end{cases} \quad (10)$$

Similarly, the complex cepstrum $\tilde{h}(m,n)$ using the MG-PS decomposition of the 2-D FSBM given by (1) can also be shown to be a simple closed-form formula of $\alpha_{m,n}$ and $\beta_{m,n}$.

Besides the above two characteristics, the 2-D FSBM possesses another characteristic as follows.

C5) As the 2-D LSI system $h(m,n)$ (with frequency response $\mathcal{H}(\omega_1,\omega_2)$) is not a 2-D FSBM (e.g., a 2-D ARMA model), the larger the chosen values for p_1 and p_2 of the 2-D FSBM $H(\omega_1,\omega_2)$, the better the approximation $H(\omega_1,\omega_2)$ to the true system $\mathcal{H}(\omega_1,\omega_2)$.

Let us conclude this section with the advantages of the proposed 2-D FSBM as follows. By C3), the stability issue is never existent since the 2-D FSBM is always stable. By C4), the calculation of the complex cepstrum of the 2-D FSBM is very simple and straightforward without the need for phase unwrapping and polynomial rooting that must be performed for the 2-D ARMA model, and therefore, it is suitable for applications using 2-D complex cepstra of signals. Complex cepstra of speech signals with the vocal tract-filter modeled as a minimum-phase AR model have been widely used in speech recognition and speaker identification [16]–[18]. Similarly, the 2-D FSBM can also be used for modeling of texture images [8], and meanwhile, its complex cepstrum obtained by (8) can be used as features for classification of texture images that will be presented later (in Section V).

III. Estimation of 2-D FSBM Parameters

Assume that $x(m,n)$ is a stationary random field that can be modeled as

$$x(m,n) = u(m,n)*h(m,n)+w(m,n) = \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty} h(i,j)u(m-i,n-j)+w(m,n) \quad (11)$$

with the following assumptions for the 2-D LSI system $h(m,n)$, the driving input $u(m,n)$ and the Gaussian noise $w(m,n)$.

A1) $h(m,n)$ is a 2-D FSBM given by (1) or (5) with p_1 and p_2 known in advance.

A2) $u(m,n)$ is a real, zero-mean, stationary, i.i.d., non-Gaussian 2-D random field with variance σ_u^2 and Mth-order ($M\geq 3$) cumulant $C_M\{u(m,n)\}\neq 0$.

A3) $w(m,n)$ is a real, zero-mean, stationary, (white or colored) Gaussian 2-D random field with variance σ_w^2.

With a given set of measurements $x(m,n), m = 0,1,\ldots,N-1, n = 0,1,\ldots,N-1$, we desire to estimate the amplitude parameters α_{i_1,i_2} and phase parameters β_{i_1,i_2} of the 2-D FSBM for all $(i_1,i_2)\in\Omega(p_1,p_2)$. Next, let us present the estimation of amplitude parameters α_{i_1,i_2} followed by the estimation of phase parameters β_{i_1,i_2}.

A. Estimation of Amplitude Parameters

The estimation of the amplitude parameters $\alpha_{i_1,i_2}, (i_1,i_2)\in\Omega(p_1,p_2)$ is equivalent to the estimation of the minimum-phase FSBM $H_{\rm MP}(\omega_1,\omega_2)$ given by (6). The minimum-phase FSBM $H_{\rm MP}(\omega_1,\omega_2)$ can be estimated using SOS-based 2-D linear prediction error (LPE) filters.

Let $v_{\rm MP}(m,n)$ be a 2-D IIR filter with the region of support $\Omega(\infty,\infty)\cup(0,0)$, and let

$$e(m,n) = x(m,n)*v_{\rm MP}(m,n) = x(m,n)+\sum\sum_{(i_1,i_2)\in\Omega(\infty,\infty)} v_{\rm MP}(i_1,i_2)x(m-i_1,n-i_2) \quad (12)$$

be the output of the filter with the input $x(m,n)$, where $v_{\rm MP}(0,0) = 1$. The optimum Wiener filter $v_{\rm MP}(m,n)$ by minimizing $E[e^2(m,n)]$ is the well-known minimum-phase 2-D LPE filter of infinite length [14], [15]. Specifically, let us model the 2-D LPE filter $v_{\rm MP}(m,n)$ as a 2-D minimum-phase FSBM as

$$V_{\rm MP}(\omega_1,\omega_2) = \exp\left\{\sum\sum_{(i_1,i_2)\in\Omega(p_1,p_2)} \tilde{\alpha}_{i_1,i_2} e^{-j(i_1\omega_1+i_2\omega_2)}\right\}. \quad (13)$$

The optimum LPE filter $\hat{V}_{\rm MP}(\omega_1,\omega_2)$ by minimizing

$$J_{\rm MSE}(\tilde{\alpha}) = E[e^2(m,n)] \quad (14)$$

where $\tilde{\alpha}$ is a vector consisting of $\tilde{\alpha}_{i_1,i_2}, \forall(i_1,i_2)\in\Omega(p_1,p_2)$, is described in the following theorem, which is 2-D extension of Theorem 1 reported in [13].

Theorem 1: Assume that $x(m,n)$ is a stationary random field given by (11) satisfying the Assumptions A1) and A2) in the absence of noise. Let $e(m,n)$ be the prediction error given by (12), where the LPE filter $V_{\rm MP}(\omega_1,\omega_2)$ is a 2-D minimum-phase FSBM given by (13). Then, $J_{\rm MSE}(\tilde{\alpha})$ defined by (14) is minimum if and only if

$$\hat{V}_{\rm MP}(\omega_1,\omega_2) = 1/H_{\rm MP}(\omega_1,\omega_2) \quad (15)$$

i.e., $\tilde{\alpha}_{i_1,i_2} = -\alpha_{i_1,i_2}, \forall(i_1,i_2)\in\Omega(p_1,p_2)$ and $\min\{J_{\rm MSE}(\tilde{\alpha})\} = \sigma_u^2$.

Based on Theorem 1, the following algorithm is proposed for the estimation of α_{i_1,i_2}.

Algorithm 1: Estimation of α_{i_1,i_2}:

Find the optimum $\hat{\tilde{\alpha}}_{i_1,i_2}$ by minimizing $J_{\rm MSE}(\tilde{\alpha})$ given by (14). Then, obtain $\hat{\alpha}_{i_1,i_2} = -\hat{\tilde{\alpha}}_{i_1,i_2}, \forall(i_1,i_2)\in\Omega(p_1,p_2)$, i.e., $\hat{H}_{\rm MP}(\omega_1,\omega_2) = 1/\hat{V}_{\rm MP}(\omega_1,\omega_2)$.

Next, let us discuss how to find the optimum $\tilde{\alpha}$ using Algorithm 1. Because $J_{\rm MSE}(\tilde{\alpha})$ is a highly nonlinear function of $\tilde{\alpha}$, it is not possible to find a closed-form solution for the optimum $\tilde{\alpha}$. Therefore, one has to resort to gradient-type iterative optimization algorithms, such as iterative Fletcher-Powell (FP) algorithm [19].

Let us present the computation of the gradient $\partial J_{\mathrm{MSE}}(\tilde{\alpha})/\partial\tilde{\alpha}$ needed by gradient-type iterative optimization algorithms. By (12) and (13), we obtain

$$\begin{aligned}\frac{\partial e(m,n)}{\partial\tilde{\alpha}_{i_1,i_2}} &= \frac{1}{(2\pi)^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} X(\omega_1,\omega_2) \\ &\quad\times\left\{\frac{\partial V_{\mathrm{MP}}(\omega_1,\omega_2)}{\partial\tilde{\alpha}_{i_1,i_2}}\right\} e^{j\omega_1 m + j\omega_2 n}\, d\omega_1\, d\omega_2 \\ &= \frac{1}{(2\pi)^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} X(\omega_1,\omega_2)V_{\mathrm{MP}}(\omega_1,\omega_2) \\ &\quad\times e^{j\omega_1(m-i_1)+j\omega_2(n-i_2)}\, d\omega_1\, d\omega_2 \\ &= e(m-i_1, n-i_2), \quad (i_1,i_2)\in\Omega(p_1,p_2).\end{aligned} \tag{16}$$

By (14) and (16), we further obtain

$$\begin{aligned}\frac{\partial J_{\mathrm{MSE}}(\tilde{\alpha})}{\partial\tilde{\alpha}_{i_1,i_2}} &= 2E\left[e(m,n)\frac{\partial e(m,n)}{\partial\tilde{\alpha}_{i_1,i_2}}\right] \\ &= 2E[e(m,n)e(m-i_1,n-i_2)] \\ &\qquad (i_1,i_2)\in\Omega(p_1,p_2).\end{aligned} \tag{17}$$

Moreover, one can see from (16) and (17) that $\partial J_{\mathrm{MSE}}(\tilde{\alpha})/\partial\tilde{\alpha}$ depends only on $e(m,n)$ and that the local minimum $J_{\mathrm{MSE}}(\tilde{\alpha})$ occurs when

$$E[e(m,n)e(m-i_1,n-i_2)] = 0, \quad (i_1,i_2)\in\Omega(p_1,p_2). \tag{18}$$

Let us conclude the amplitude parameter estimation with the following two remarks.

R1) The optimum prediction error $e(m,n)$ that is orthogonal to $e(m-i_1,n-i_2)$ for all $(i_1,i_2)\in\Omega(p_1,p_2)$ by (18) is a 2-D white random field as

$$e(m,n) = x(m,n) * \hat{v}_{\mathrm{MP}}(m,n) = u(m,n) * h_{\mathrm{AP}}(m,n). \tag{19}$$

In other words, $\hat{v}_{\mathrm{MP}}(m,n)$ is a 2-D whitening filter, and $e(m,n)$ is an amplitude equalized signal with a flat power spectral density equal to σ_u^2.

R2) When the 2-D LSI system $h(m,n)$ is a 2-D FSBM with unknown p_1 and p_2, the obtained $\hat{H}_{\mathrm{MP}}(\omega_1,\omega_2)$ is merely an approximation to $H_{\mathrm{MP}}(\omega_1,\omega_2)$ if the chosen values for p_1 and p_2 in (13) are smaller than the true values of p_1 and p_2. This implies that as the 2-D LSI system $h(m,n)$ is not a 2-D FSBM, the larger the chosen values for p_1 and p_2 in (13), the better the approximation $\hat{H}_{\mathrm{MP}}(\omega_1,\omega_2)$ to the minimum-phase system associated with $h(m,n)$.

B. Estimation of Phase Parameters

The estimation of the phase parameters $\beta_{i_1,i_2}, (i_1,i_2)\in\Omega(p_1,p_2)$ is equivalent to the estimation of the 2-D allpass FSBM $H_{\mathrm{PS}}(\omega_1,\omega_2)$ given by (3) from $x(m,n)$, as well as equivalent to the estimation of the 2-D allpass FSBM $H_{\mathrm{AP}}(\omega_1,\omega_2)$ given by (7) from the amplitude equalized signal $e(m,n)$ given by (19). The estimation of both $H_{\mathrm{PS}}(\omega_1,\omega_2)$ and $H_{\mathrm{AP}}(\omega_1,\omega_2)$ are based on the following theorem proposed by Chien *et al.* [20] for phase equalization using HOS.

Theorem 2 [Theorem 1 in [20]]: Assume that $\bar{x}(m,n) = u(m,n) * \bar{h}(m,n)$, where $u(m,n)$ satisfies the assumption A2), and $\bar{h}(m,n)$ is a real stable 2-D LSI system. Let $\bar{y}(m,n) = \bar{x}(m,n) * v_{\mathrm{AP}}(m,n)$, where $v_{\mathrm{AP}}(m,n)$ is a 2-D allpass filter. Then, the absolute Mth-order ($M\geq 3$) cumulant $|C_M\{\bar{y}(m,n)\}|$ of $\bar{y}(m,n)$ is maximized if and only if

$$\arg\{V_{\mathrm{AP}}(\omega_1,\omega_2)\} = -\arg\{\bar{H}(\omega_1,\omega_2)\} + \tau_1\omega_1 + \tau_2\omega_2 \tag{20}$$

where τ_1 and τ_2 are unknown integers.

Specifically, let $v_{\mathrm{AP}}(m,n)$ be a 2-D allpass FSBM given by

$$V_{\mathrm{AP}}(\omega_1,\omega_2) = \exp\left\{ j \sum\sum_{(i_1,i_2)\in\Omega(p_1,p_2)} \gamma_{i_1,i_2}\sin(i_1\omega_1 + i_2\omega_2)\right\} \tag{21}$$

and let γ be a vector consisting of $\gamma_{i_1,i_2}, \forall(i_1,i_2)\in\Omega(p_1,p_2)$. Let

$$y_1(m,n) = e(m,n) * v_{\mathrm{AP}}(m,n) \tag{22}$$

$$y_2(m,n) = x(m,n) * v_{\mathrm{AP}}(m,n) \tag{23}$$

where $e(m,n)$ is the amplitude equalized signal given by (19), and

$$J_{\mathrm{CUM}}(\gamma) = |C_M\{y(m,n)\}| \tag{24}$$

where $M\geq 3$ and $y(m,n) = y_1(m,n)$ or $y(m,n) = y_2(m,n)$. Then, we have the following two facts, assuming that $h(m,n)$ in (11) is a 2-D FSBM given by (1) or (5) with p_1 and p_2 known in advance.

F1) For $y(m,n) = y_1(m,n)$, $J_{\mathrm{CUM}}(\gamma)$ is maximized if and only if $\arg\{V_{\mathrm{AP}}(\omega_1,\omega_2)\} = -\arg\{H_{\mathrm{AP}}(\omega_1,\omega_2)\}$, i.e., $\gamma_{i_1,i_2} = -(\alpha_{i_1,i_2}+\beta_{i_1,i_2}), \forall(i_1,i_2)\in\Omega(p_1,p_2)$.

F2) For $y(m,n) = y_2(m,n)$, $J_{\mathrm{CUM}}(\gamma)$ is maximized if and only if $\arg\{V_{\mathrm{AP}}(\omega_1,\omega_2)\} = -\arg\{H_{\mathrm{PS}}(\omega_1,\omega_2)\}$, i.e., $\gamma_{i_1,i_2} = -\beta_{i_1,i_2}, \forall(i_1,i_2)\in\Omega(p_1,p_2)$.

Because the proofs of F1) and F2) are quite similar, let us only prove F2) as follows.

Proof of F2): Since the HOS of Gaussian processes (due to $w(m,n)$ in (11)) are equal to zero, the Gaussian noise $w(m,n)$ is negligible in the following proof. By Theorem 2, (1), and (21), we have

$$\begin{aligned}&\arg\{V_{\mathrm{AP}}(\omega_1,\omega_2)\} + \arg\{H_{\mathrm{PS}}(\omega_1,\omega_2)\} \\ &\quad= \sum\sum_{(i_1,i_2)\in\Omega(p_1,p_2)} (\gamma_{i_1,i_2}+\beta_{i_1,i_2})\sin(i_1\omega_1+i_2\omega_2) \\ &\qquad\qquad |\omega_1|\leq\pi, \quad |\omega_2|\leq\pi \\ &\quad= \tau_1\omega_1+\tau_2\omega_2, \quad |\omega_1|\leq\pi, \quad |\omega_2|\leq\pi \\ &\quad= \sum\sum_{(i_1,i_2)\in\Omega(\infty,\infty)} b_{i_1,i_2}\sin(i_1\omega_1+i_2\omega_2) \\ &\qquad\qquad |\omega_1|\leq\pi, \quad |\omega_2|\leq\pi\end{aligned} \tag{25}$$

where τ_1 and τ_2 are integers, and

$$b_{i_1,i_2} = \begin{cases}\frac{2\tau_1}{i_1}(-1)^{i_1+1}, & i_2 = 0 \text{ and } i_1 > 0 \\ \frac{2\tau_2}{i_2}(-1)^{i_2+1}, & i_1 = 0 \text{ and } i_2 > 0 \\ 0, & \text{otherwise}\end{cases} \tag{26}$$

are coefficients of the Fourier series expansion of the sum of the two linear functions $\tau_1\omega_1$ and $\tau_2\omega_2$. From the second and fourth lines of (25), one can see that

$$b_{i_1,i_2} = 0, \quad \forall(i_1,i_2)\in\Omega(\infty,\infty) - \Omega(p_1,p_2) \tag{27}$$

which, together with (26), leads to $\tau_1 = \tau_2 = 0$, i.e.,

$$b_{i_1,i_2} = 0, \quad \forall (i_1,i_2) \in \Omega(\infty,\infty). \tag{28}$$

Therefore, from (25) and (28), one can obtain $\gamma_{i_1,i_2} = -\beta_{i_1,i_2}, \forall (i_1,i_2) \in \Omega(p_1,p_2)$. □

Next, let us present the two algorithms for estimating the parameters of the 2-D FSBM based on F1) and F2), respectively.

Algorithm 2: Estimation of β_{i_1,i_2} based on F1):

S1) Find the optimum $\hat{\gamma}$ by maximizing $J_{\rm CUM}(\gamma)$ defined by (24) with $y(m,n) = y_1(m,n)$.

S2) Obtain $\hat{H}_{\rm AP}(\omega_1,\omega_2)$ by $\hat{H}_{\rm AP}(\omega_1,\omega_2) = 1/\hat{V}_{\rm AP}(\omega_1,\omega_2)$ and $\hat{\beta}_{i_1,i_2} = -\hat{\gamma}_{i_1,i_2} - \hat{\alpha}_{i_1,i_2}, \forall (i_1,i_2) \in \Omega(p_1,p_2)$, where $\hat{\alpha}_{i_1,i_2}, \forall (i_1,i_2) \in \Omega(p_1,p_2)$ are the amplitude parameter estimates obtained by Algorithm 1.

Algorithm 3: Estimation of β_{i_1,i_2} based on F2):

S1) Find the optimum $\hat{\gamma}$ by maximizing $J_{\rm CUM}(\gamma)$ defined by (24) with $y(m,n) = y_2(m,n)$.

S2) Obtain $\hat{H}_{\rm PS}(\omega_1,\omega_2)$ by $\hat{H}_{\rm PS}(\omega_1,\omega_2) = 1/\hat{V}_{\rm AP}(\omega_1,\omega_2)$ and $\hat{\beta}_{i_1,i_2} = -\hat{\gamma}_{i_1,i_2}, \forall (i_1,i_2) \in \Omega(p_1,p_2)$.

As $J_{\rm MSE}(\tilde{\alpha})$ used by Algorithm 1, the objective function $J_{\rm CUM}(\gamma)$ is also a highly nonlinear function of parameters γ_{i_1,i_2}. For instance, $J_{\rm CUM}(\gamma)$ is given by

$$J_{\rm CUM}(\gamma) = \begin{cases} |E[y^3(m,n)]|, & \text{for } M=3 \\ |E[y^4(m,n)] - 3\{E[y^2(m,n)]\}^2|, & \text{for } M=4. \end{cases} \tag{29}$$

Therefore, we also need resort to the gradient-type iterative optimization algorithms such as the iterative FP algorithm for finding a local maximum of $J_{\rm CUM}(\gamma)$ that needs

$$\frac{\partial y(m,n)}{\partial \gamma_{i_1,i_2}} = \frac{1}{2}\{y(m+i_1,n+i_2) - y(m-i_1,n-i_2)\} \tag{30}$$

to compute the gradient $\partial J_{\rm CUM}(\gamma)/\partial\gamma$. The proof of (30) is similar to that of (16) and, thus, is omitted here.

Four worthy remarks regarding the proposed Algorithms 2 and 3 are as follows.

R3) Prior to using Algorithm 2, the amplitude parameter estimates $\hat{\alpha}_{i_1,i_2}$ of the 2-D FSBM, and the optimum prediction error $e(m,n)$ must be obtained using Algorithm 1. This is not required by Algorithm 3.

R4) When the LSI system $h(m,n)$ is not a 2-D FSBM, the unknown linear phase terms $\omega_1\tau_1 + \omega_2\tau_2$ may affect the resultant estimates $\hat{H}_{\rm AP}(\omega_1,\omega_2)$ and $\hat{H}_{\rm PS}(\omega_1,\omega_2)$. This can be easily verified from (25) in the proof of F2). Algorithm 3 may well end up with the optimum $\hat{\gamma}_{i_1,i_2} = -\beta_{i_1,i_2} + b_{i_1,i_2}, \forall (i_1,i_2) \in \Omega(p_1,p_2)$ for p_1 and p_2 chosen sufficiently large, leading to a 2-D space shift in the resultant estimate $\hat{h}(m+\tau_1, n+\tau_2)$. This can happen in using Algorithm 2 as well.

R5) By (19) and F1), the optimum phase equalized signal $y_1(m,n)$ obtained by Algorithm 2 is also a deconvolved signal using the inverse filter $\hat{H}_{\rm INV}(\omega_1,\omega_2) = 1/\hat{H}(\omega_1,\omega_2)$, i.e.,

$$\begin{aligned} y_1(m,n) &= e(m,n) * \hat{v}_{\rm AP}(m,n) \\ &= x(m,n) * \hat{h}_{\rm INV}(m,n) \simeq u(m,n). \end{aligned} \tag{31}$$

R6) When the 2-D LSI system $h(m,n)$ is a 2-D FSBM with unknown p_1 and p_2, the obtained estimates $\hat{H}_{\rm AP}(\omega_1,\omega_2)$ and $\hat{H}_{\rm PS}(\omega_1,\omega_2)$ are merely an approximation to $H_{\rm AP}(\omega_1,\omega_2)$ and $H_{\rm PS}(\omega_1,\omega_2)$, respectively, if the chosen values for p_1 and p_2 in (21) are smaller than the true values of p_1 and p_2.

The optimum estimate $\hat{H}(\omega_1,\omega_2)$ can be obtained either using Algorithms 1 and 2 or using Algorithms 1 and 3. However, when the 2-D LSI system $h(m,n)$ (with frequency response $\mathcal{H}(\omega_1,\omega_2)$) is not a 2-D FSBM, the larger the chosen values for p_1 and p_2, the better the approximation $\hat{H}(\omega_1,\omega_2)$ to the true system $\mathcal{H}(\omega_1,\omega_2)$, except for an unknown linear phase (a 2-D space shift), as mentioned in C5) and R4).

Finally, let us conclude this subsection with a discussion for the computational complexity of the proposed Algorithms 1, 2, and 3. In practice, the second-order cumulant used by Algorithm 1 [see (14)] and the higher order cumulants used by Algorithms 2 and 3 [see (24)] must be replaced with the associated sample cumulants. The computation of both $e(m,n)$ for the former and $y(m,n)$ for the latter can be efficiently performed using 2-D FFT because the 2-D FSBM is a parametric model in frequency domain. Moreover, both $\partial e(m,n)/\partial\tilde{\alpha}_{i_1,i_2}$ [see (16)] for the former and $\partial y(m,n)/\partial\gamma_{i_1,i_2}$ [see (30)] for the latter have a parallel structure suitable for software and hardware implementation of the three proposed algorithms. Next, let us investigate the consistency of the three proposed estimators.

C. *Consistency of the Proposed Estimators*

Recall that the second- and higher order cumulants used in Algorithms 1, 2, and 3 must be replaced by the associated sample cumulants in practice. Let $\xi(m,n)$ be a 2-D $N \times N$ zero-mean stationary non-Gaussian signal and $\hat{C}_{\mathcal{M}}\{\xi(m,n)\}$ ($\mathcal{M} \geq 2$) be the associated sample cumulant of the $\mathcal{M}$th-order cumulant of $\xi(m,n)$. For instance

$$\hat{C}_2\{\xi(m,n)\} = \frac{1}{N^2}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\xi^2(m,n) \tag{32}$$

$$\hat{C}_3\{\xi(m,n)\} = \frac{1}{N^2}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\xi^3(m,n) \tag{33}$$

$$\hat{C}_4\{\xi(m,n)\} = \frac{1}{N^2}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\xi^4(m,n) - 3(\hat{C}_2\{\xi(m,n)\})^2. \tag{34}$$

The consistency of the three proposed estimators can be proved following the same procedure as the proof of [7, Prop. 3] with the following proposition used.

Proposition 1 [Theorem 2a in [21]]: As $N \to \infty$

$$\hat{C}_{\mathcal{M}}\{\xi(m,n)\} \overset{\rm w.p.1}{\to} C_{\mathcal{M}}\{\xi(m,n)\} \tag{35}$$

where $\mathcal{M} \geq 2$ and $a \overset{\rm w.p.1}{\to} b$ denotes "a converges to b with probability one." □

First of all, let us prove that $\hat{\alpha}_{i_1,i_2}$ obtained by Algorithm 1 are consistent estimates. By Proposition 1 and (14), one can easily infer that

$$\begin{aligned} \hat{J}_{\rm MSE}(\tilde{\alpha}) = \hat{C}_2\{e(m,n)\} &\overset{\rm w.p.1}{\to} J_{\rm MSE}(\tilde{\alpha}) \\ &= E[e^2(m,n)] \quad \text{uniformly in } \tilde{\alpha} \end{aligned} \tag{36}$$

as $N \to \infty$, which implies that the optimum $\hat{\tilde{\alpha}}$ by minimizing $\hat{J}_{\text{MSE}}(\tilde{\alpha})$ converges to the optimum $\tilde{\alpha}$ by minimizing $J_{\text{MSE}}(\tilde{\alpha})$ with probability one as $N \to \infty$. Therefore, by Algorithm 1 and Theorem 1, one can infer that

$$\hat{\alpha}_{i_1,i_2} = -\hat{\tilde{\alpha}}_{i_1,i_2} \overset{\text{w.p.1}}{\to} -\tilde{\alpha}_{i_1,i_2} = \alpha_{i_1,i_2}, \quad \forall (i_1,i_2) \in \Omega(p_1,p_2) \tag{37}$$

as $N \to \infty$.

Next, let us prove the consistency of the phase parameter estimates $\hat{\beta}_{i_1,i_2}$ obtained by Algorithm 2. By Proposition 1 and (24), it can be easily inferred that

$$\begin{aligned}\hat{J}_{\text{CUM}}(\gamma) &= |\hat{C}_M\{y_1(m,n)\}| \overset{\text{w.p.1}}{\to} J_{\text{CUM}}(\gamma) \\ &= |C_M\{y_1(m,n)\}| \quad \text{uniformly in } \gamma\end{aligned} \tag{38}$$

as $N \to \infty$. Equation (38) also implies that the optimum $\hat{\gamma}$ by maximizing $\hat{J}_{\text{CUM}}(\gamma)$ converges to the optimum γ by maximizing $J_{\text{CUM}}(\gamma)$ with probability one as $N \to \infty$. Therefore, by Algorithm 2, F1), and (37), we can infer that

$$\hat{\beta}_{i_1,i_2} = -\hat{\gamma}_{i_1,i_2} - \hat{\alpha}_{i_1,i_2} \overset{\text{w.p.1}}{\to} -\gamma_{i_1,i_2} - \alpha_{i_1,i_2} = \beta_{i_1,i_2} \quad \forall (i_1,i_2) \in \Omega(p_1,p_2) \tag{39}$$

as $N \to \infty$.

The consistency of Algorithm 3 can be proved similarly as we did for Algorithm 2 above and, thus, is omitted here.

IV. Simulation Results

In this section, three simulation examples are to be presented to justify the efficacy of the three proposed algorithms for the estimation of the 2-D FSBM parameters. In the three examples, the driving input signal $u(m,n)$ was a zero-mean, exponentially distributed, i.i.d., random field with variance $\sigma_u^2 = 1$ that was convolved with a 2-D LSI system followed by addition of white Gaussian noise to generate the synthetic $N \times N$ data $x(m,n)$. Then the three proposed algorithms are used to process $x(m,n)$. The iterative FP algorithm was employed to obtain the optimum amplitude parameters α_{i_1,i_2} by Algorithm 1 and phase parameters β_{i_1,i_2} by Algorithms 2 and 3, respectively, with the cumulant order $M = 3$. The initial condition used for the FP algorithm is a zero vector (associated with $\alpha_{i_1,i_2} = 0, \forall (i_1,i_2)$ and $\beta_{i_1,i_2} = 0, \forall (i_1,i_2)$). Thirty independent runs were performed in each of the three examples.

Example 1 considers the case that the true system $h(m,n)$ is a 2-D FSBM. Examples 2 and 3 consider the case that $h(m,n)$ is a 2-D MA model and the case that $h(m,n)$ is a 2-D ARMA model, respectively. In Example 1, mean and root mean square error (RMSE) of the obtained 30 amplitude and phase parameter estimates were calculated. In Examples 2 and 3, a normalized MSE (NMSE) [22] defined as

$$\text{NMSE} = \frac{\frac{1}{30}\sum_{i=1}^{30}\sum_{m=-15}^{15}\sum_{n=-15}^{15}(h(m,n) - \hat{h}_i(m,n))^2}{\sum_{m=-15}^{15}\sum_{n=-15}^{15} h^2(m,n)} \tag{40}$$

was calculated, where $\hat{h}_i(m,n)$ (normalized with the same energy as $h(m,n)$) is the estimate $\hat{h}(m,n)$ obtained in the ith run, and the time delay between $\hat{h}_i(m,n)$ and the true $h(m,n)$ was artificially removed. Next, let us turn to Example 1.

TABLE I
Simulation Results for Example 1. The True 2-D FSBM Parameters, Mean, and RMSE of 30 Independent Estimates of the 2-D FSBM Parameters Obtained by Algorithms 1, 2 and 3, Respectively

FSBM of $H(\omega_1,\omega_2)$	FSBM of $\hat{H}(\omega_1,\omega_2)$ Algorithm 1		FSBM of $H(\omega_1,\omega_2)$	FSBM of $\hat{H}(\omega_1,\omega_2)$ Algorithm 2		FSBM of $\hat{H}(\omega_1,\omega_2)$ Algorithm 3	
	mean	RMSE		mean	RMSE	mean	RMSE
$\alpha_{0,1} = -0.97$	−0.9696	0.0068	$\beta_{0,1} = -0.27$	−0.2717	0.0126	−0.2703	0.0175
$\alpha_{1,-1} = -0.5$	−0.4987	0.0085	$\beta_{1,-1} = -0.95$	−0.9526	0.0103	−0.9496	0.0230
$\alpha_{1,0} = 1.04$	1.0394	0.0089	$\beta_{1,0} = 0.18$	0.1784	0.0110	0.1774	0.0356
$\alpha_{1,1} = 0.52$	0.5199	0.0101	$\beta_{1,1} = -0.92$	−0.9232	0.0123	−0.9304	0.0231

TABLE II
Simulation Results for Example 2. NMSE of 30 Independent Channel Estimates Obtained by Algorithms 1 and 2 and Algorithms 1 and 3, Respectively

	Algorithms 1 & 2				Algorithms 1 & 3			
	SNR (dB)							
$p_1 (= p_2)$	20	15	10	5	20	15	10	5
1	0.0614	0.0603	0.0610	0.0760	0.0593	0.0584	0.0594	0.0750
2	0.0077	0.0083	0.0122	0.0309	0.0078	0.0084	0.0122	0.0306
3	0.0043	0.0052	0.0097	0.0307	0.0047	0.0054	0.0098	0.0303

Example 1—Estimation of the 2-D FSBM: A 2-D FSBM $H(\omega_1,\omega_2)$ given by (1) with parameters

$$\alpha_{0,1} = -0.97 \quad \alpha_{1,-1} = -0.5 \quad \alpha_{1,0} = 1.04 \quad \alpha_{1,1} = 0.52$$
$$\beta_{0,1} = -0.27 \quad \beta_{1,-1} = -0.95 \quad \beta_{1,0} = 0.18 \quad \beta_{1,1} = -0.92$$

$(p_1 = p_2 = 1)$ was used in this example. The simulation results for $N = 128$ and SNR $= \infty$ are shown in Table I. One can see, from this table, that mean values of $\hat{\alpha}_{i_1,i_2}$ and $\hat{\beta}_{i_1,i_2}$ are close to the true values of α_{i_1,i_2} and β_{i_1,i_2}, respectively, and the associated RMSEs are also small. These simulation results support that the three proposed algorithms are effective for the estimation of the 2-D FSBM.

Example 2—Approximation to MA Model: A 2-D MA system taken from [20] was used in this example. The system input-output relation is given by

$$\begin{aligned}x(m,n) = {} & u(m,n) - 0.8u(m-1,n) + 0.2u(m-2,n) \\ & + 1.8u(m,n-1) - 1.44u(m-1,n-1) \\ & + 0.36u(m-2,n-1) \\ & - 0.5u(m,n-2) + 0.4u(m-1,n-2) \\ & - 0.1u(m-2,n-2) \\ & + 0.5u(m,n-3) - 0.4u(m-1,n-3) \\ & + 0.1u(m-2,n-3).\end{aligned} \tag{41}$$

Table II shows the simulation results (NMSEs) obtained by Algorithms 1 and 2 and Algorithms 1 and 3 for $N = 128$, and $(p_1,p_2) = (1,1), (2,2), (3,3)$ and SNR $= 5, 10, 15, 20$ dB, respectively. One can see from this table that NMSEs are small and decrease as SNR or p_1 $(=p_2)$ increases. These simulation results support that the 2-D FSBM estimates obtained by Algorithms 1 and 2 and Algorithms 1 and 3 are good approximations to the true 2-D MA system.

TABLE III
SIMULATION RESULTS FOR EXAMPLE 3. NMSE OF 30 INDEPENDENT CHANNEL ESTIMATES OBTAINED BY ALGORITHMS 1 AND 2 AND ALGORITHMS 1 AND 3, RESPECTIVELY

	Algorithms 1 & 2				Algorithms 1 & 3			
	SNR (dB)							
$p_1\,(=p_2)$	20	15	10	5	20	15	10	5
2	0.1471	0.1224	0.1268	0.1850	0.0676	0.0712	0.0899	0.1545
3	0.0352	0.0361	0.0482	0.1058	0.0294	0.0308	0.0431	0.0949
4	0.0244	0.0275	0.0373	0.0830	0.0232	0.0243	0.0345	0.0810

Example 3—Approximation to ARMA Model: A 2-D ARMA system with a nonsymmetric support taken from [6] was used in this example. The system input–output relation is given by

$$\begin{aligned} x(m,n) &- 0.004x(m+1,n+1) + 0.0407x(m+1,n) \\ &- 0.027x(m+1,n-1) - 0.2497x(m,n+1) \\ &- 0.568x(m,n-1) + 0.1037x(m-1,n+1) \\ &- 0.3328x(m-1,n) + 0.1483x(m-1,n-1) \\ &= u(m,n) - 0.5u(m+1,n) - 0.5u(m,n+1) \\ &\quad - u(m,n-1) - u(m-1,n). \end{aligned} \tag{42}$$

Table III shows the simulation results of NMSEs obtained by Algorithms 1 and 2 and Algorithms 1 and 3 for $N = 128$, and $(p_1, p_2) = (2,2), (3,3), (4,4)$ and SNR $= 5, 10, 15, 20$ dB, respectively. One can see from this table that NMSEs are smaller for larger SNR or larger p_1 $(=p_2)$ as $p_1 \geq 3$. These simulation results support that the 2-D FSBM estimates obtained by Algorithms 1 and 2 and Algorithms 1 and 3 are good approximations to the true 2-D ARMA system as p_1 and p_2 are sufficient.

V. TEXTURE IMAGE CLASSIFICATION USING THE 2-D FSBM

This section considers the application of the 2-D FSBM to texture image classification because a texture image can be modeled as a 2-D non-Gaussian random field given by (11) in the absence of noise [8]. For comparison, the feature vector, which is denoted by θ_1, exploiting the higher order statistical features of texture images proposed by Tsatsanis and Giannakis [9] and the feature vector, which is denoted by θ_2, using toroidal lattice simultaneous AR (SAR) model parameters proposed by Kashyap and Chellappa [2] were also employed for texture image classification. Next, let us briefly present the feature vectors θ_1 and θ_2, respectively.

Assume that the $N \times N$ texture image $x(m,n)$ can be modeled as the output signal of a noncausal ARMA $(\boldsymbol{k}, \boldsymbol{l})$ system driven by a zero-mean, i.i.d. non-Gaussian input signal $u(m,n)$, where $\boldsymbol{k} = [k_1, k_2]$ is the order of the AR part, and $\boldsymbol{l} = [l_1, l_2]$ is the order of the MA part. Let

$$\begin{aligned} &C_{3x}([m_1,n_1],[m_2,n_2]) \\ &= E[x(m,n)x(m+m_1,n+n_1)x(m+m_2,n+n_2)] \end{aligned} \tag{43}$$

denote the third-order cumulant of $x(m,n)$ with lags $([m_1,n_1],[m_2,n_2])$. Tsatsanis and Giannakis [9] considered the following feature vector:

$$\theta_1 = \{\hat{C}_{3x}([m_1,n_1],[m_2,n_2]), \forall([m_1,n_1],[m_2,n_2]) \in \Omega_1\} \tag{44}$$

for classification of texture images where

$$\begin{aligned} &\hat{C}_{3x}([m_1,n_1],[m_2,n_2]) \\ &= \frac{1}{N^2}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} x(m,n) \\ &\quad \times x(m+m_1,n+n_1)x(m+m_2,n+n_2) \end{aligned} \tag{45}$$

(third-order sample cumulant) and

$$\Omega_1 \triangleq \{([m_1,n_1],[m_2,n_2]) : |m_i|, |n_i| \leq l_j + 3k_j \quad i=1,2, j=1,2\}. \tag{46}$$

Thanks to the asymptotic Gaussianity of cumulant estimates, Tsatsanis and Giannakis [9] proposed an asymptotic ML classifier (in the cumulant domain) by maximizing

$$\begin{aligned} &J_{\mathrm{ML}}^{(c)}(\theta_1) \\ &= -\frac{1}{2}\left\{\frac{1}{N^2}\log|\boldsymbol{\Sigma}_c| + [\theta_1 - \bar{\theta}_{c,1}]^T(\boldsymbol{\Sigma}_c)^{-1}[\theta_1 - \bar{\theta}_{c,1}]\right\} \end{aligned} \tag{47}$$

where $\bar{\theta}_{c,1}$ and $\boldsymbol{\Sigma}_c, c = 1, 2, \ldots, \mathcal{C}$ (number of total classes) denote the mean and asymptotic covariance matrix of θ_1 associated with class c. Note that as $\boldsymbol{\Sigma}_c$ is an identity matrix, the ML criterion reduces to a minimum Euclidean distance (MED) criterion, i.e.,

$$J_{\mathrm{MED}}^{(c)}(\theta_1) = ||\theta_1 - \bar{\theta}_{c,1}||^2 \tag{48}$$

where $||\theta||$ denotes the Euclidean norm of θ. The asymptotic ML classifier would gain statistical efficiency over the MED classifier at the expense of higher computational complexity. Note that $\bar{\theta}_{c,1}$ and $\boldsymbol{\Sigma}_c$ can be estimated during the training phase of the classifier.

On the other hand, the SAR model [2] for the $N \times N$ texture image $x(m,n)$ with region of support $\mathcal{R} = \{(m,n) : m = 0, 1, \ldots, N-1, n = 0, 1, \ldots, N-1\}$ is

$$x(m,n) = \sum\sum_{(i_1,i_2)\in\Omega_2(p_1,p_2)} a_{i_1,i_2} x(m \oplus i_1, n \oplus i_2) + w(m,n) \quad (m,n) \in \mathcal{R} \tag{49}$$

where

- $\oplus$ addition modulo N operator;
- Ω_2 region of support for AR parameters a_{i_1,i_2};
- $w(m,n)$ white Gaussian with variance σ_w^2.

Kashyap and Chellappa [2] estimate $\hat{a}_{i_1,i_2}$ and $\hat{\sigma_w^2}$ using an AML algorithm to form the feature vector

$$\theta_2 = \left[\hat{\boldsymbol{a}}^T, \widehat{\sigma_w^2}/\sigma_x^2\right]^T \tag{50}$$

where $\hat{\boldsymbol{a}}$ is a vector consisting of AR parameters estimates $\hat{a}_{i_1,i_2}, \forall(i_1,i_2) \in \Omega_2(p_1,p_2)$, and σ_x^2 is the variance of $x(m,n)$. Note that as Ω_2 is symmetric, the AR parameters are symmetric (i.e., $a_{i_1,i_2} = a_{-i_1,-i_2}$), as presented in [2], which implies that the 2-D AR model is zero-phase in this case.

Next, two new feature vectors, which are denoted by θ_3 and θ_4, based on the 2-D FSBM, are considered for texture image classification. They are defined as

$$\theta_3 = \left[\hat{\alpha}^T, \sigma_e^2/\sigma_x^2\right]^T \tag{51}$$

$$\theta_4 = [\hat{\alpha}^T, C_M\{y(m,n)\}/(E[y^2(m,n)])^{M/2}]^T \tag{52}$$

where $\hat{\alpha}$ is the obtained amplitude parameter vector of the 2-D FSBM using Algorithm 1, and $y(m,n) = y_1(m,n)$ [see (31)] obtained by Algorithm 2 or

$$y(m,n) = x(m,n) * \hat{h}_{\mathrm{INV}}(m,n) \tag{53}$$

where $\hat{h}_{\mathrm{INV}}(m,n)$ is the inverse system of the 2-D FSBM $\hat{H}(\omega_1,\omega_2)$ obtained using Algorithms 1 and 3. Note that the second component of θ_4 is nothing but the normalized Mth-order $(M \geq 3)$ cumulant of $y(m,n)$, which is invariant for $\lambda y(m-\tau_1, n-\tau_2)$ for any integers τ_1 and τ_2 and any nonzero λ. Next, let us present why θ_3 and θ_4 can be used for texture image classification.

As mentioned in Section II, complex cepstra of speech signals with the vocal tract-filter modeled as a minimum-phase AR model have been widely used in speech recognition and speaker identification. This motivates the application of amplitude parameters α of the 2-D FSBM, i.e., minimum-phase parameters in the MP-AP decomposition, to texture image classification simply because α and the complex cepstrum $\tilde{h}_{\mathrm{MP}}(m,n)$ given by (9) are the same. However, the phase parameters β of the 2-D FSBM cannot be used because of unknown 2-D space shift (τ_1,τ_2), as mentioned in R4). Nevertheless, the deconvolved signal $y(m,n)$ approximates an i.i.d. non-Gaussian random field characterized by HOS such as normalized higher order cumulants. Next, let us present some experimental results using the proposed feature vectors θ_3 and θ_4.

The texture images used for classification were taken from University of Southern California—Signal and Image Processing Institute (USC-SIPI) Image Data Base. Twelve 512×512 texture images were chosen for classification, including grass, treebark, straw, herringbone, wool, leather, water, wood, raffia, brickwall, plastic, and sand. Each image was divided into 16 128×128 nonoverlapping subimages to provide 12 classes of 16 subimages each. For each subimage, θ_1 was obtained using (45) by including 31 nonredundant third-order cumulants in the set $\{C_{3x}([m_1,n_1],[m_2,n_2]), m_1 = 0, n_1 = 0,1, 0 \leq m_2, n_2 \leq 3, ([m_1,n_1],[m_2,n_2]) \neq ([0,0],[0,1])\}$ (since $C_{3x}([0,0],[0,1]) = C_{3x}([0,1],[0,0])$ is redundant), θ_2 was obtained using the AML algorithm with $\Omega_2 = \{(i,j) : i = -2,\ldots,2, j = -2,\ldots,2, (i,j) \neq (0,0)\}$, including 13 nonredundant entries, and θ_3 and θ_4 were obtained with $p_1 = p_2 = 2$ and $p_1 = p_2 = 3$, respectively. Note that either of θ_3 and θ_4 includes 13 and 25 entries for $p_1 = p_2 = 2$ and $p_1 = p_2 = 3$, respectively. The iterative FP algorithm with the initial condition set to a zero vector [associated with $\alpha_{i_1,i_2} = 0, \forall (i_1,i_2)$ and $\beta_{i_1,i_2} = 0, \forall (i_1,i_2)$] was used to obtain θ_3 and θ_4 by the three proposed algorithms.

The leave-one-out strategy [1] was then used to perform the classification. To perform classification with the chosen subimage of a specific class c, the mean feature vector $\bar{\theta}_{c,i}$ (associated with $\theta_i, i = 1,2,3,4$) and covariance matrix $\mathbf{\Sigma}_c$ (which is only needed by θ_1 with the ML criterion used) was calculated from the other 15 subimages of the class c, whereas for the other 11 classes, the $\bar{\theta}_{\tilde{c},i}$ and $\mathbf{\Sigma}_{\tilde{c}}$ $(\tilde{c} \neq c)$ were calculated from all 16 subimages of each class. The classification procedure was repeated for $16 \times 12 = 192$ subimages. The number of misclassifications out of 192 classification operations is used as the performance index.

TABLE IV
EXPERIMENTAL RESULTS USING FEATURE VECTOR θ_1

Texture size: 128×128, Misclassifications: 43 of 192												
Texture	1	2	3	4	5	6	7	8	9	10	11	12
1. grass	**10**	0	**1**	0	0	0	0	0	0	**3**	0	**2**
2. treebark	0	**15**	0	0	0	0	0	0	0	**1**	0	0
3. straw	**2**	0	**10**	0	0	0	0	0	0	**1**	0	**3**
4. herringbone	0	0	0	**16**	0	0	0	0	0	0	0	0
5. wool	0	0	0	0	**9**	0	**7**	0	0	0	0	0
6. leather	0	0	0	0	0	**12**	0	0	0	0	**4**	0
7. water	0	0	0	0	**5**	0	**11**	0	0	0	0	0
8. wood	0	0	0	0	0	0	0	**16**	0	0	0	0
9. raffia	0	0	0	0	0	0	0	0	**16**	0	0	0
10. brickwall	**2**	**2**	0	**2**	0	0	0	0	0	**9**	0	**1**
11. plastic	0	0	0	0	**1**	**1**	**4**	0	**1**	0	**9**	0
12. sand	0	0	0	0	0	0	0	0	0	0	0	**16**

TABLE V
EXPERIMENTAL RESULTS USING FEATURE VECTOR θ_2

Texture size: 128×128, Misclassifications: 31 of 192												
Texture	1	2	3	4	5	6	7	8	9	10	11	12
1. grass	**13**	0	0	0	0	**1**	0	0	0	0	**2**	0
2. treebark	0	**15**	0	0	0	0	0	0	**1**	0	0	0
3. straw	**2**	0	**12**	0	0	**1**	0	0	0	0	0	**1**
4. herringbone	0	0	0	**16**	0	0	0	0	0	0	0	0
5. wool	0	0	0	**4**	**12**	0	0	0	0	0	0	0
6. leather	0	0	0	**4**	0	**12**	0	0	0	0	0	0
7. water	0	0	0	0	0	0	**10**	**2**	**1**	**3**	0	0
8. wood	0	0	0	0	0	0	0	**16**	0	0	0	0
9. raffia	0	0	0	0	0	0	0	0	**16**	0	0	0
10. brickwall	0	0	0	**3**	0	0	0	0	**1**	**12**	0	0
11. plastic	0	**3**	0	0	0	0	0	0	0	0	**11**	**2**
12. sand	0	0	0	0	0	0	0	0	0	0	0	**16**

Recall that the MED criterion $J_{\mathrm{MED}}^{(c)}$ applies to the classifier using any one of $\theta_i, i = 1,2,3,4$, whereas the ML criterion $J_{\mathrm{ML}}^{(c)}$ is only applicable as θ_1 is used. For obtaining reliable $\bar{\theta}_{c,1}$ (31×1 vector) and $\mathbf{\Sigma}_c$ (31×31 matrix) for $c = 1,2,\ldots,\mathcal{C} = 12$, we further divided each 128×128 subimage into four 64×64 nonoverlapping sub-subimages to obtain a larger sample space. Then, $\bar{\theta}_{c,1}$ and $\mathbf{\Sigma}_c$ of the class c were calculated from the other 15 subimages $(N = 128)$ and the associated 60 sub-subimages $(N = 64)$ of the class c, whereas for each of the other 11 classes, $\bar{\theta}_{\tilde{c},1}$ and $\mathbf{\Sigma}_{\tilde{c}}$ $(\tilde{c} \neq c)$ were calculated from all the 16 subimages $(N = 128)$ and the associated 64 sub-subimages $(N = 64)$.

The classification results associated with the MED classifier using $\theta_i, i = 1,2,3,4$ are shown in Tables IV through XV, respectively. Tables IV and V show the classification results using feature vectors θ_1 and θ_2, respectively. Table VI shows the classification results using feature vector θ_3 for $p_1 = p_2 = 2$. Tables VII and VIII show the classification results using feature vector θ_4 for $p_1 = p_2 = 2$ associated with Algorithms 1 and 2 for the cumulant order $M = 3$ and $M = 4$, respectively, and the corresponding results associated with Algorithms 1 and 3 for $M = 3$ and $M = 4$ are shown in Tables IX and X, respectively. The results corresponding to those shown in Tables VI–X for

TABLE VI
EXPERIMENTAL RESULTS USING FEATURE VECTOR θ_3 ASSOCIATED WITH ALGORITHM 1 FOR $p_1 = p_2 = 2$

Texture size: 128×128, Misclassifications: 10 of 192												
Texture	1	2	3	4	5	6	7	8	9	10	11	12
1. grass	14	0	0	0	0	1	0	0	0	0	0	1
2. treebark	0	16	0	0	0	0	0	0	0	0	0	0
3. straw	2	0	13	0	0	0	0	0	0	0	0	1
4. herringbone	0	0	0	16	0	0	0	0	0	0	0	0
5. wool	0	0	0	0	15	0	0	0	0	0	1	0
6. leather	0	0	0	0	0	16	0	0	0	0	0	0
7. water	0	0	0	0	0	0	13	0	0	3	0	0
8. wood	0	0	0	0	0	0	0	16	0	0	0	0
9. raffia	0	0	0	0	0	0	0	0	16	0	0	0
10. brickwall	0	1	0	0	0	0	0	0	0	15	0	0
11. plastic	0	0	0	0	0	0	0	0	0	0	16	0
12. sand	0	0	0	0	0	0	0	0	0	0	0	16

TABLE VII
EXPERIMENTAL RESULTS USING FEATURE VECTOR θ_4 ASSOCIATED WITH ALGORITHMS 1 AND 2 FOR $M = 3$ AND $p_1 = p_2 = 2$

Texture size: 128×128, Misclassifications: 7 of 192												
Texture	1	2	3	4	5	6	7	8	9	10	11	12
1. grass	15	0	0	0	0	1	0	0	0	0	0	0
2. treebark	0	16	0	0	0	0	0	0	0	0	0	0
3. straw	1	1	13	1	0	0	0	0	0	0	0	0
4. herringbone	0	0	0	16	0	0	0	0	0	0	0	0
5. wool	0	0	0	0	16	0	0	0	0	0	0	0
6. leather	0	0	0	0	0	16	0	0	0	0	0	0
7. water	0	0	0	0	0	0	13	0	0	3	0	0
8. wood	0	0	0	0	0	0	0	16	0	0	0	0
9. raffia	0	0	0	0	0	0	0	0	16	0	0	0
10. brickwall	0	0	0	0	0	0	0	0	0	16	0	0
11. plastic	0	0	0	0	0	0	0	0	0	0	16	0
12. sand	0	0	0	0	0	0	0	0	0	0	0	16

TABLE VIII
EXPERIMENTAL RESULTS USING FEATURE VECTOR θ_4 ASSOCIATED WITH ALGORITHMS 1 & 2 FOR $M = 4$ AND $p_1 = p_2 = 2$

Texture size: 128×128, Misclassifications: 8 of 192												
Texture	1	2	3	4	5	6	7	8	9	10	11	12
1. grass	15	0	0	0	0	1	0	0	0	0	0	0
2. treebark	0	16	0	0	0	0	0	0	0	0	0	0
3. straw	1	0	13	1	0	1	0	0	0	0	0	0
4. herringbone	0	0	0	16	0	0	0	0	0	0	0	0
5. wool	0	0	0	0	16	0	0	0	0	0	0	0
6. leather	0	0	0	0	0	16	0	0	0	0	0	0
7. water	0	0	0	0	0	0	12	0	0	3	0	1
8. wood	0	0	0	0	0	0	0	16	0	0	0	0
9. raffia	0	0	0	0	0	0	0	0	16	0	0	0
10. brickwall	0	0	0	0	0	0	0	0	0	16	0	0
11. plastic	0	0	0	0	0	0	0	0	0	0	16	0
12. sand	0	0	0	0	0	0	0	0	0	0	0	16

$p_1 = p_2 = 3$ are shown in Tables XI–XV, respectively. Each row of these tables includes correct classifications (diagonal term) and some misclassifications (off diagonal terms) over the performed 16 subimage classifications of the associated class.

TABLE IX
EXPERIMENTAL RESULTS USING FEATURE VECTOR θ_4 ASSOCIATED WITH ALGORITHMS 1 AND 3 FOR $M = 3$ AND $p_1 = p_2 = 2$

Texture size: 128×128, Misclassifications: 8 of 192												
Texture	1	2	3	4	5	6	7	8	9	10	11	12
1. grass	15	0	0	0	0	1	0	0	0	0	0	0
2. treebark	0	16	0	0	0	0	0	0	0	0	0	0
3. straw	2	0	13	1	0	0	0	0	0	0	0	0
4. herringbone	0	0	0	16	0	0	0	0	0	0	0	0
5. wool	0	0	0	0	16	0	0	0	0	0	0	0
6. leather	0	0	0	0	0	16	0	0	0	0	0	0
7. water	0	0	0	0	0	0	13	0	0	3	0	0
8. wood	0	0	0	0	0	0	0	16	0	0	0	0
9. raffia	0	0	0	0	0	0	0	0	16	0	0	0
10. brickwall	0	0	0	0	0	0	0	0	0	16	0	0
11. plastic	0	1	0	0	0	0	0	0	0	0	15	0
12. sand	0	0	0	0	0	0	0	0	0	0	0	16

TABLE X
EXPERIMENTAL RESULTS USING FEATURE VECTOR θ_4 ASSOCIATED WITH ALGORITHMS 1 AND 3 FOR $M = 4$ AND $p_1 = p_2 = 2$

Texture size: 128×128, Misclassifications: 8 of 192												
Texture	1	2	3	4	5	6	7	8	9	10	11	12
1. grass	15	0	0	0	0	1	0	0	0	0	0	0
2. treebark	0	16	0	0	0	0	0	0	0	0	0	0
3. straw	1	0	13	0	0	1	0	0	0	0	0	1
4. herringbone	0	0	0	16	0	0	0	0	0	0	0	0
5. wool	0	0	0	0	16	0	0	0	0	0	0	0
6. leather	0	0	0	0	0	16	0	0	0	0	0	0
7. water	0	0	0	0	0	0	13	0	0	3	0	0
8. wood	0	0	0	0	0	0	0	16	0	0	0	0
9. raffia	0	0	0	0	0	0	0	0	16	0	0	0
10. brickwall	0	0	0	0	0	0	0	0	0	16	0	0
11. plastic	0	1	0	0	0	0	0	0	0	0	15	0
12. sand	0	0	0	0	0	0	0	0	0	0	0	16

TABLE XI
EXPERIMENTAL RESULTS USING FEATURE VECTOR θ_3 ASSOCIATED WITH ALGORITHM 1 FOR $p_1 = p_2 = 3$

Texture size: 128×128, Misclassifications: 5 of 192												
Texture	1	2	3	4	5	6	7	8	9	10	11	12
1. grass	16	0	0	0	0	0	0	0	0	0	0	0
2. treebark	0	16	0	0	0	0	0	0	0	0	0	0
3. straw	1	0	13	0	0	1	0	0	0	0	0	1
4. herringbone	0	0	0	16	0	0	0	0	0	0	0	0
5. wool	0	0	0	0	16	0	0	0	0	0	0	0
6. leather	0	0	0	0	0	16	0	0	0	0	0	0
7. water	0	0	0	0	0	0	15	1	0	0	0	0
8. wood	0	0	0	0	0	0	0	16	0	0	0	0
9. raffia	0	0	0	0	0	0	0	0	16	0	0	0
10. brickwall	0	0	0	0	0	0	0	0	0	16	0	0
11. plastic	0	1	0	0	0	0	0	0	0	0	15	0
12. sand	0	0	0	0	0	0	0	0	0	0	0	16

The MED classifier using θ_1 and θ_2 yielded 43 (Table IV) and 31 (Table V) misclassifications, respectively. The one using θ_3 yielded ten (Table VI) and five (Table XI) misclassifications for $p_1 = p_2 = 2$ and $p_1 = p_2 = 3$, respectively. The one using

TABLE XII
EXPERIMENTAL RESULTS USING FEATURE VECTOR θ_4 ASSOCIATED WITH ALGORITHMS 1 AND 2 FOR $M = 3$ AND $p_1 = p_2 = 3$

Texture size: 128×128, Misclassifications: 4 of 192												
Texture	1	2	3	4	5	6	7	8	9	10	11	12
1. grass	**16**	0	0	0	0	0	0	0	0	0	0	0
2. treebark	0	**16**	0	0	0	0	0	0	0	0	0	0
3. straw	**1**	0	**14**	0	0	**1**	0	0	0	0	0	0
4. herringbone	0	0	0	**16**	0	0	0	0	0	0	0	0
5. wool	0	0	0	0	**16**	0	0	0	0	0	0	0
6. leather	0	0	0	0	0	**16**	0	0	0	0	0	0
7. water	0	0	0	0	0	0	**14**	**1**	0	**1**	0	0
8. wood	0	0	0	0	0	0	0	**16**	0	0	0	0
9. raffia	0	0	0	0	0	0	0	0	**16**	0	0	0
10. brickwall	0	0	0	0	0	0	0	0	0	**16**	0	0
11. plastic	0	0	0	0	0	0	0	0	0	0	**16**	0
12. sand	0	0	0	0	0	0	0	0	0	0	0	**16**

TABLE XIII
EXPERIMENTAL RESULTS USING FEATURE VECTOR θ_4 ASSOCIATED WITH ALGORITHMS 1 AND 2 FOR $M = 4$ AND $p_1 = p_2 = 3$

Texture size: 128×128, Misclassifications: 6 of 192												
Texture	1	2	3	4	5	6	7	8	9	10	11	12
1. grass	**15**	0	0	0	0	0	0	0	0	0	0	**1**
2. treebark	0	**16**	0	0	0	0	0	0	0	0	0	0
3. straw	**1**	0	**13**	0	0	**1**	0	0	0	0	0	**1**
4. herringbone	0	0	0	**16**	0	0	0	0	0	0	0	0
5. wool	0	0	0	0	**16**	0	0	0	0	0	0	0
6. leather	0	0	0	0	0	**16**	0	0	0	0	0	0
7. water	0	0	0	0	0	0	**15**	**1**	0	0	0	0
8. wood	0	0	0	0	0	0	0	**16**	0	0	0	0
9. raffia	0	0	0	0	0	0	0	0	**16**	0	0	0
10. brickwall	0	0	0	0	0	0	0	0	0	**16**	0	0
11. plastic	0	**1**	0	0	0	0	0	0	0	0	**15**	0
12. sand	0	0	0	0	0	0	0	0	0	0	0	**16**

TABLE XIV
EXPERIMENTAL RESULTS USING FEATURE VECTOR θ_4 ASSOCIATED WITH ALGORITHMS 1 AND 3 FOR $M = 3$ AND $p_1 = p_2 = 3$

Texture size: 128×128, Misclassifications: 4 of 192												
Texture	1	2	3	4	5	6	7	8	9	10	11	12
1. grass	**16**	0	0	0	0	0	0	0	0	0	0	0
2. treebark	0	**16**	0	0	0	0	0	0	0	0	0	0
3. straw	**1**	0	**13**	0	0	**1**	0	0	0	0	0	**1**
4. herringbone	0	0	0	**16**	0	0	0	0	0	0	0	0
5. wool	0	0	0	0	**16**	0	0	0	0	0	0	0
6. leather	0	0	0	0	0	**16**	0	0	0	0	0	0
7. water	0	0	0	0	0	0	**15**	**1**	0	0	0	0
8. wood	0	0	0	0	0	0	0	**16**	0	0	0	0
9. raffia	0	0	0	0	0	0	0	0	**16**	0	0	0
10. brickwall	0	0	0	0	0	0	0	0	0	**16**	0	0
11. plastic	0	0	0	0	0	0	0	0	0	0	**16**	0
12. sand	0	0	0	0	0	0	0	0	0	0	0	**16**

θ_4 yielded seven or eight misclassifications (Tables VII-X) and four to six misclassifications (Tables XII–XV) for $p_1 = p_2 = 2$ and $p_1 = p_2 = 3$, respectively. Some noteworthy observations

TABLE XV
EXPERIMENTAL RESULTS USING FEATURE VECTOR θ_4 ASSOCIATED WITH ALGORITHMS 1 AND 3 FOR $M = 4$ AND $p_1 = p_2 = 3$

Texture size: 128×128, Misclassifications: 4 of 192												
Texture	1	2	3	4	5	6	7	8	9	10	11	12
1. grass	**16**	0	0	0	0	0	0	0	0	0	0	0
2. treebark	0	**16**	0	0	0	0	0	0	0	0	0	0
3. straw	**1**	0	**13**	0	0	**1**	0	0	0	0	0	**1**
4. herringbone	0	0	0	**16**	0	0	0	0	0	0	0	0
5. wool	0	0	0	0	**16**	0	0	0	0	0	0	0
6. leather	0	0	0	0	0	**16**	0	0	0	0	0	0
7. water	0	0	0	0	0	0	**16**	0	0	0	0	0
8. wood	0	0	0	0	0	0	0	**16**	0	0	0	0
9. raffia	0	0	0	0	0	0	0	0	**16**	0	0	0
10. brickwall	0	0	0	0	0	0	0	0	0	**16**	0	0
11. plastic	0	**1**	0	0	0	0	0	0	0	0	**15**	0
12. sand	0	0	0	0	0	0	0	0	0	0	0	**16**

from Tables IV–XV are as follows. The MED classifier using either of θ_3 and θ_4 performs better for larger $p_1 = p_2$. The MED classifier using θ_4 performs slightly better than the one using θ_3, and both of them perform much better than the one using either of θ_1 and θ_2. These experimental results support that the proposed feature vectors θ_3 and θ_4 are effective for texture image classification. However, θ_3 (without using higher order cumulants) seems sufficient for this application.

Besides the results shown in Table IV associated with the MED classifier using the feature vector θ_1, the ML classifier using θ_1 was also tested with the same texture images where the first term in $J_{\rm ML}^{(c)}$ [see (47)] was ignored since it is negligible [9]. This classifier achieved zero misclassification (perfect classification) (with no need of showing the results by table) over the 192 classification operations. Because $\mathbf{\Sigma}_c$ is a 31×31 matrix, the computational complexity associated with $J_{\rm ML}^{(c)}$ is much higher than that associated with $J_{\rm MED}^{(c)}$ [see (48)] during the training phase and operation phase.

VI. CONCLUSION

Chi's 1-D FSBM has been extended to the 2-D FSBM [see (1) and (5)] that can be used as an approximation (with stability guarantee) to an arbitrary 2-D LSI system, and its complex cepstrum can be easily obtained from its amplitude and phase parameters [see (8)–(10)] with no need of complicated 2-D phase unwrapping and polynomial rooting. Then, Algorithm 1 was presented for amplitude parameter estimation, and Algorithms 2 and 3 were presented for phase parameter estimation, followed by the establishment of their consistency. Some simulation results were provided to support that the three proposed algorithms are effective for the estimation of the 2-D FSBM parameters. Then, two new feature vectors [see (51) and (52)] obtained by the three proposed algorithms were presented for texture image classification followed by some experimental results for demonstrating their efficacy. However, the determination of (p_1, p_2) of the 2-D FSBM is left for future research. Other applications of the proposed 2-D FSBM such as texture image synthesis are also left for future research.

REFERENCES

[1] R. L. Kashyap, R. Chellappa, and A. Khotanzad, "Texture classification using features derived from random field models," *Pattern Recognit. Lett.*, vol. 1, no. 1, pp. 43–50, 1982.

[2] R. L. Kashyap and R. Chellappa, "Estimation and choice of neighbors in spatial-interaction models of images," *IEEE Trans. Inform. Theory*, vol. IT-29, pp. 58–72, Jan. 1983.

[3] R. Chellappa and S. Chatterjee, "Classification of texture using Gaussian Markov random fields," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 959–963, Apr. 1985.

[4] K. B. Eom, "2-D moving average models for texture synthesis and analysis," *IEEE Trans. Image Processing*, vol. 7, pp. 1741–1746, Dec. 1998.

[5] C. L. Nikias and A. P. Petropulu, *Higher-Order Spectra Analysis*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[6] J. K. Tugnait, "Estimation of linear parametric models of nonGaussian discrete random fields with application to texture synthesis," *IEEE Trans. Image Processing*, vol. 3, pp. 109–127, Mar. 1994.

[7] T. E. Hall and G. B. Giannakis, "Image modeling using inverse filtering criteria with application to textures," *IEEE Trans. Image Processing*, vol. 5, pp. 938–949, June 1996.

[8] ——, "Bispectral analysis and model validation of texture images," *IEEE Trans. Image Processing*, vol. 4, pp. 996–1009, July 1995.

[9] M. K. Tsatsanis and G. B. Giannakis, "Object and texture classification using higher order statistics," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 14, pp. 733–750, July 1992.

[10] C.-Y. Chi and C.-H. Chen, "Two-dimensional frequency domain blind system identification using higher order statistics with application to texture synthesis," *IEEE Trans. Signal Processing*, vol. 49, pp. 864–877, Apr. 2001.

[11] A. Swami, G. B. Giannakis, and J. M. Mendel, "Linear modeling of multidimensional non-Gaussian processes using cumulants," *Multidimen. Syst. Signal Process.*, vol. 1, pp. 11–37, 1990.

[12] C.-Y. Chi, "Fourier series based nonminimum phase model for second- and higher-order statistical signal processing," *Proc. IEEE Signal Process. Workshop Higher-Order Statist.*, pp. 395–399, July 21–23, 1997.

[13] ——, "Fourier series based nonminimum phase model for statistical signal processing," *IEEE Trans. Signal Processing*, vol. 47, pp. 2228–2240, Aug. 1999.

[14] D. E. Dudgeon and R. M. Mersereau, *Multidimensional Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1984.

[15] S. M. Kay, *Modern Spectral Estimation*. Englewood Cliffs, NJ: Prentice-Hall, 1988.

[16] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*. Englewood Cliffs, NJ: Prentice-Hall, 1978.

[17] L. R. Rabiner, K. C. Pan, and F. K. Soong, "On the performance of isolated word speech recognizers using vector quantization and temporal energy contours," *AT&T Bell Labs Tech. J.*, vol. 63, no. 7, pp. 1245–1260, 1984.

[18] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[19] D. M. Burley, *Studies in Optimization*. New York: Wiley, 1974.

[20] H.-M. Chien, H.-L. Yang, and C.-Y. Chi, "Parametric cumulant based phase estimation of 1-D and 2-D nonminimum phase systems by allpass filtering," *IEEE Trans. Signal Processing*, vol. 45, pp. 1742–1762, July 1997.

[21] G. B. Giannakis and M. K. Tsatanis, "A unifying maximum-likelihood view of cumulant and polyspectral measures for non-Gaussian signal classification and estimation," *IEEE Trans. Inform. Theory*, vol. 38, pp. 386–406, Mar. 1992.

[22] J. K. Tugnait, "Identification and deconvolution of multichannel linear non-Gaussian processes using higher order statistics and inverse filter criteria," *IEEE Trans. Signal Processing*, vol. 45, pp. 658–672, Mar. 1997.

[23] A. Swami, G. B. Giannakis, and G. Zhou, "Bibliography on higher-order statistics," *Signal Process.*, vol. 60, no. 1, pp. 65–126, July 1997.

Chii-Horng Chen was born in Taiwan, R.O.C., on December 2, 1970. He received the B.S. degree from the Department of Control Engineering, National Chiao Tung University, Hsinchu, Taiwan, in 1992 and Ph.D. degree from the Department of Electrical Engineering, National Tsing Hua University, Hsinchu, in 2001.

Since October 2001, he has been with ADMtek Incorporated, Hsinchu, where he works on the analysis, design, and implementation of the transceivers for wireless local area networks. His research interests include wireless communications, statistical signal processing, and digital signal processing.

Chong-Yung Chi (S'83–M'83–SM'89) was born in Taiwan, R.O.C., on August 7, 1952. He received the B.S. degree from the Tatung Institute of Technology, Taipei, Taiwan, in 1975, the M.S. degree from the National Taiwan University (NTU), Taipei, in 1977, and the Ph.D. degree from the University of Southern California, Los Angeles, in 1983, all in electrical engineering.

From July 1983 to September 1988, he was with the Jet Propulsion Laboratory, Pasadena, CA, where he worked on the design of various spaceborne radar remote sensing systems including radar scatterometers, synthetic aperture radars, altimeters, and rain mapping radars. From October 1988 to July 1989, he was a visiting specialist at the Department of Electrical Engineering, NTU. Since August 1989, he has been a Professor with the Department of Electrical Engineering, National Tsing Hua University, Hsinchu, Taiwan. He was a visiting researcher at the Advanced Telecommunications Research (ATR) Institute International, Kyoto, Japan, in May and June 2001. He has published more than 100 technical papers on radar remote sensing, system identification and estimation theory, deconvolution and channel equalization, digital filter design, spectral estimation, and higher order statistics (HOS)-based signal processing. He has been a reviewer for both *Signal Processing* and *Electronics Letters*. His research interests include signal processing for wireless communications, statistical signal processing, and digital signal processing and their applications.

Dr. Chi is an active member of Society of Exploration Geophysicists, a member of European Association for Signal Processing, and an active member of the Chinese Institute of Electrical Engineering. He was technical committee member of both the 1997 and 1999 IEEE Signal Processing Workshop on Higher Order Statistics (HOS) and the 2001 IEEE Workshop on Statistical Signal Processing (SSP). He is also a member of International Advisory Committee of TENCON 2001. He is a co-organizer and general co-chairman of 2001 IEEE SP Workshop on Signal Processing Advances in Wireless Communications (SPAWC-2001). Since 1983, he has served as a reviewer for several international journals and conferences, such as the IEEE TRANSACTIONS ON SIGNAL PROCESSING, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, and the IEEE TRANSACTIONS ON COMMUNICATIONS. Currently, he is an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING, and Chair of the Information Theory Chapter of the IEEE Taipei Section.

Ching-Yung Chen was born in Taiwan, R.O.C., on November 14, 1975. He received the B.S. degree from the Department of Electrical Engineering, National Tsing Hua University (NTHU), Hsinchu, Taiwan, in 1997, and is currently pursuing the Ph.D. degree at NTHU.

His research interests include statistical signal processing, digital signal processing, and wireless communications.