Analysing textures with exponential analysis

Deepayan Bhowmik, Yuan Hou, Annie Cuyt and Wen-shin Lee

***Abstract*—A majority of the visual pattern recognition tasks involve texture analysis within its processing pipeline. One of the key steps for texture analysis is the frequency decomposition of the input signal. We present a new approach using a recent 2-dimensional exponential analysis technique that can provide a basis for several vision algorithms. Exponential analysis (EA) offers the advantage of sparsity in the model and continuity in the parameters. This results in a much more compact representation of textures when compared to traditional Fourier or wavelet transform techniques, especially when the texture patterns are homogeneous and follow an underlying real or complex exponential model.**

In support of our proposed method, extensive experiments are conducted using synthetic as well as real texture images from standard benchmark datasets. The results outperform the Discrete Fourier Transform (DFT) in representing texture patterns, with significantly fewer terms, while retaining RMSE values after reconstruction. To demonstrate the usefulness of our method, we develop a new fabric defect detection algorithm which exhibits very acceptable performance measures. Additional demonstration of a texture classification method illustrates the potential for a wider adaptation of the proposed texture decomposition approach.

***Index Terms*—Exponential analysis, multivariate, texture decomposition, fabric defect detection, texture classification.**

I. Introduction

Frequency decomposition is a fundamental but challenging inverse problem to most image and signal processing applications. Such applications include visual pattern recognition tasks that involve texture analysis using frequency decomposition within its pipeline to extract meaningful features. Major frequency decomposition approaches can be categorised into 1) template based convolution methods, *e.g.*, Fourier [1], cosine [2] or wavelet transforms [3], [4] and 2) data driven adaptive approaches, *e.g.*, empirical mode decomposition (EMD) [5], [6] or empirical wavelet transform (EWT) [7], [8]. Template based transforms are considered to be rigid and rely on predetermined basis functions or frames that are agnostic of the input image. On the contrary, adaptive techniques are flexible and can provide a better representation of the data. One common challenge across all these methods is the model cardinality of the frequency domain representation, which is often dictated by the available data granularity. Also, the mentioned techniques do not exploit in any way the structure present in the available data, such as in texture data. While fine grained transform domain representations are required for image reconstruction purposes, it is not always necessary for many applications, such as in texture pattern recognition. This paper proposes a new image decomposition technique using the recent multivariate exponential analysis of [9], with the aim to decompose texture images using a minimal representation.

In the past few years, multidimensional exponential analysis has attracted considerable attention in computational mathematics as well as in signal processing. In the 1-dimensional case, the Prony-like exponential analysis methods, such as matrix pencil [10], ESPRIT [11], TLS-Prony [12], MUSIC [13] have all been successfully applied in solving many practical problems. At the same time, several multi-dimensional versions of these Prony-like methods have been developed, *e.g.*, [14]–[19]. However, due to complexity issues, until recently these methods were not very suitable to serve as a general tool for higher-dimensional decomposition.

In [9], a d-dimensional exponential model of n terms can be recovered from $O((d{+}1)n)$ regularly collected samples, which is substantially less than with other multi-dimensional Prony-like methods, where both the sample usage and computational complexity explode exponentially. This opens a wealth of possibilities, including certain image processing applications. Texture is a fundamental component of any image and is encountered in most image analysis problems. Therefore, it is no surprise that it is a very intensively researched area [20].

This paper explores the use of the multivariate exponential analysis presented in [9] as a new image decomposition technique that can express regular texture patterns with substantially fewer parameters. The key focus of this work constitutes of the mathematical formulation of a new decomposition technique validated on both synthetic and real images available from benchmark data sets. Usefulness of the proposed approach is demonstrated by a new fabric defect detection algorithm. Further usage scenarios include a texture classification application, illustrating the potential for a wider applicability. The main contributions of our work are:

- Formulation of multivariate exponential analysis as a new image decomposition tool for texture analysis,
- Sparse image representation and reconstruction using only a limited number of terms,
- Development of a new fabric defect detection algorithm that uses the proposed decomposition tool, and
- Demonstration of exponential analysis in the case of texture classification.

The initial concept and early results of this approach were reported as a conference publication [21]. This work extends it by presenting the details of our approach, discusses extended results, and proposes an updated fabric defect detection algorithm with performance comparison against the state-of-the-art.

D. Bhowmik and W.-s. Lee are with the Division of Computing Science and Mathematics at the University of Stirling, Stirling FK9 4LA, United Kingdom. (e-mail: {deepayan.bhowmik, wen-shin.lee}@stir.ac.uk).

Y. Hou and A. Cuyt are with the Department of Computer Science, University of Antwerp, 2020 Antwerp, Belgium. A. Cuyt is also with the College of Mathematics and Statistics, Shenzhen University, Shenzhen, Guangdong 518060, China. (e-mail: {yuan.hou, annie.cuyt}@uantwerpen.be).

Manuscript received XXX XX, XXXX; revised XXX XX, XXXX; accepted for publication XXX XX, XXXX;.

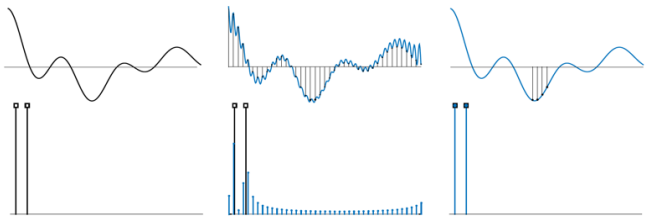


Fig. 1: DFT versus EA of a bi-exponential signal. Top left to right: original signal, DFT reconstruction, EA reconstruction. Bottom left to right: frequency content of original, DFT frequency analysis, EA frequency analysis. EA recovers this signal from only 4 measurements and provides the two significant frequencies.

II. Preliminaries

Exponential functions are one of the fundamental objects in linear and non-linear models for extracting relevant structures from measured data. The objective in basic univariate multi-exponential analysis is to recover the coefficients α_j and the exponents ϕ_j in the parametric multi-exponential sum

$$f(t) = \sum_{j=1}^{n} \alpha_j \exp(\phi_j t), \qquad \alpha_j, \phi_j \in \mathbb{C} \tag{1}$$

from a limited number of samples of $f(t)$. In the ideal noise-free situation this number is known to be a mere $2n$. Often, also n is unknown and needs to be determined on the fly. Many generalisations of the basic problem statement exist. As illustrated in Fig. 1, the approach of EA compared to that of the DFT is quite distinct although both use exponential functions as a building block for their spectral analysis:

- the DFT frequency resolution is a function of the sampling duration, so of time,
- and its frequency set is a predetermined discrete grid, thus causing leakage when ϕ_j does not match the grid,
- and its frequency set is a predetermined discrete grid, thus causing leakage when ϕ_j does not match the grid,
- while the number of required samples in EA depends on the sparsity n of $f(t)$,
- and the method supports continuity in the spectral analysis output for ϕ_j.

Let us denote the imaginary part of a complex number by $\Im(\cdot)$ and let $f(t)$ be sampled equidistantly at $k\Delta, k = 0, 1, 2, \ldots$ with

$$|\Im(\phi_j)\Delta| < \pi, \qquad j = 1, \ldots, n$$

in order to comply with the Nyquist sampling constraint. Let $H_n^{(m)}$ denote the $n \times n$ Hankel matrix

$$H_n^{(m)} = \begin{pmatrix} f_m & \cdots & f_{m+n-1} \\ \vdots & \iddots & \vdots \\ f_{m+n-1} & \cdots & f_{m+2n-2} \end{pmatrix}, \qquad f_k = f(k\Delta).$$

We further introduce $E_j = \exp(\phi_j \Delta)$ which we assume to be distinct. How to deal with the coalescence of some of these values is discussed in [22]. Because of the inherent structure (1) which holds at each of the sample points f_k, the matrix $H_n^{(m)}$ can be factored as

$$H_n^{(m)} = V_n D_\alpha D_E^m V_n^T,$$

where V_n is the Vandermonde matrix

$$V_n = \left(E_j^{i-1}\right)_{i,j=1}^{n}$$

and D_α and D_E are diagonal matrices respectively filled with the vectors $(\alpha_1, \ldots, \alpha_n)$ and $(E_1, \ldots, E_n)$. This factorisation is easy to verify directly by performing the matrix product at the right hand side and compare the matrix entries in the product to those of the Hankel matrix. So the $E_j, j = 1, \ldots, n$ can be found as the generalized eigenvalues $\lambda_j, j = 1, \ldots, n$ of the problem [10]

$$H_n^{(1)} v_j = \lambda_j H_n^{(0)} v_j, \tag{2}$$

where the $v_j, j = 1, \ldots, n$ are the right generalized eigenvectors. Setting up (2) requires only the $2n$ samples $f_0, \ldots, f_{2n-1}$. From the generalized eigenvalues $E_j = \exp(\phi_j \Delta)$ the complex values ϕ_j can be extracted uniquely because $|\Im(\phi_j)\Delta| < \pi$. After recovering the E_j, the α_j can be computed from the Vandermonde structured linear system

$$\sum_{j=1}^{n} \alpha_j E_j^k = f_k, \qquad k = 0, \ldots, 2n-1, \ldots \tag{3}$$

In a noisefree mathematical context, only n equations of (3) are linearly independent because of the relationship (2) between the E_j. How to reliably proceed in a noisy context is analyzed in great detail in [23]. Generally speaking, when dealing with noisy data, the sparsity n is overestimated by $\eta > n$ and the minimum number of required samples 2η is again overestimated by $N > 2\eta > 2n$. The square $n \times n$ numerical linear algebra problems (2) and (3) then take the sizes $(N-\eta) \times \eta$ and $N \times \eta$ respectively and are solved in the least squares sense. Often the least squares solution of (2) is preceded by a rank reduction step of the $(N-\eta) \times \eta$ Hankel matrices, thus reducing the value of η to a smaller value before solving the overdetermined versions of (2) and (3). When n cannot be determined through other means (see [24] and [23]) then this reduced value of η takes the role of n. With respect to the conditioning of the $(N-\eta) \times \eta$ Hankel matrices, $\eta \approx N/3$ makes a good choice.

In image processing the argument $\phi_j t$ of the exponential function in (1) is replaced by the inner product $\langle \phi_j, p \rangle$ as in (4), where $\phi_j = (\phi_{jx}, \phi_{jy})$ and $p = (x, y)$ is the pixel location in the image.

III. Exponential image analysis

We approach image decomposition as a two-dimensional exponential analysis problem. That is, we seek to determine n and retrieve $\alpha_j, \phi_{jx}, \phi_{jy} \in \mathbb{C}$ from as few evaluations of

$$f(x, y) = \sum_{j=1}^{n} \alpha_j \exp(\phi_{jx} x + \phi_{jy} y), \tag{4}$$

as possible, where (x, y) is the location of a pixel and $f(x, y)$ the value at the corresponding pixel. When $f(x, y)$ is periodic such as in some textures, and can be decomposed as a linear

combination of undamped sine and cosine functions, then ϕ_{jx} and ϕ_{jy} are purely imaginary. Here we allow for more general images.

The method we present is based on the new sparse algorithms [9], [23] requiring only $O(3n)$ samples to analyze (4). We first summarize the 2-d idea explained in [9], and afterwards we explain how it is combined with the 1-d technique of [23].

A. Basic idea

Let $\Delta = (\Delta_x, \Delta_y) \neq (0,0)$ and $\delta = (\delta_x, \delta_y) \neq (0,0)$ be linearly independent, with

$$\begin{aligned} |\Im(\phi_{jx}\Delta_x + \phi_{jy}\Delta_y)| < \pi, \qquad & j = 1, \ldots, n, \\ |\Im(\phi_{jx}\delta_x + \phi_{jy}\delta_y)| < \pi, \qquad & j = 1, \ldots, n, \end{aligned} \tag{5}$$

and let the values $\exp(\phi_{jx}\Delta_x + \phi_{jy}\Delta_y), j = 1, \ldots, n$ be mutually distinct. How to deal with non-distinct values is discussed in [9] and summarized further down. We sample $f(x,y)$ at the equidistant points $k\Delta$ and some shifted locations $k\Delta + \delta$:

$$\begin{aligned} f_k &:= f(k\Delta_x, k\Delta_y), \qquad k = 0, \ldots, 2n-1, \\ F_k &:= f(k\Delta_x + \delta_x, k\Delta_y + \delta_y), \qquad k = 0, \ldots, n-1. \end{aligned}$$

Then first, the expressions $\exp(\phi_{jx}\Delta_x + \phi_{jy}\Delta_y), j = 1, \ldots, n$ are retrieved as the generalized eigenvalues λ_j of

$$\begin{pmatrix} f_1 & f_2 & \cdots & f_n \\ f_2 & \cdots & & f_{n+1} \\ \vdots & & & \vdots \\ f_n & f_{n+1} & \cdots & f_{2n-1} \end{pmatrix} v_j = \lambda_j \begin{pmatrix} f_0 & f_1 & \cdots & f_{n-1} \\ f_1 & \cdots & & f_n \\ \vdots & & & \vdots \\ f_{n-1} & f_n & \cdots & f_{2n-2} \end{pmatrix} v_j, \tag{6}$$

where the v_j denote the right eigenvectors. Several numerical methods exist for the solution of this problem, among which [10]–[12], [23] used in Section V. Because of (5) we can uniquely retrieve the inner products

$$\Phi_j := \langle \phi_j, \Delta \rangle, \qquad \phi_j = (\phi_{jx}, \phi_{jy}), \qquad j = 1, \ldots, n$$

from the computed $\lambda_j = \exp(\Phi_j)$. We're not yet able to recover the individual ϕ_{jx} and ϕ_{jy} though.

Second, we rewrite the values F_k as

$$\begin{aligned} F_k &= \sum_{j=1}^{n} \alpha_j \exp(\phi_{jx}\delta_x + \phi_{jy}\delta_y) \exp(k\Phi_j) \\ &= \sum_{j=1}^{n} A_j \exp^k(\Phi_j), \qquad A_j := \alpha_j \exp(\phi_{jx}\delta_x + \phi_{jy}\delta_y) \end{aligned}$$

and we introduce the notations $\alpha := (\alpha_1, \ldots, \alpha_n)^T, A := (A_1, \ldots, A_n)$. We solve the linear systems of interpolation conditions

$$\begin{pmatrix} 1 & \cdots & 1 \\ \exp(\Phi_1) & \cdots & \exp(\Phi_n) \\ \vdots & & \vdots \\ \exp^{2n-1}(\Phi_1) & \cdots & \exp^{2n-1}(\Phi_n) \end{pmatrix} \alpha = \begin{pmatrix} f_0 \\ \vdots \\ f_{2n-1} \end{pmatrix}, \tag{7}$$

and

$$\begin{pmatrix} 1 & \cdots & 1 \\ \exp(\Phi_1) & \cdots & \exp(\Phi_n) \\ \vdots & & \vdots \\ \exp^{n-1}(\Phi_1) & \cdots & \exp^{n-1}(\Phi_n) \end{pmatrix} A = \begin{pmatrix} F_0 \\ \vdots \\ F_{n-1} \end{pmatrix} \tag{8}$$

and define $\exp(\Psi_j) := A_j/\alpha_j = \exp(\langle \phi_j, \delta \rangle), j = 1, \ldots, n$. Note that we have no problem to pair the Ψ_j to the $\Phi_j, j = 1, \ldots, n$ since the A_j are paired to the $\alpha_j, j = 1, \ldots, n$ through the Vandermonde systems (7) and (8).

The fact that the vectors Δ and δ are linearly independent leads for each $j = 1, \ldots, n$ to the regular linear system

$$\begin{pmatrix} \Delta_x & \Delta_y \\ \delta_x & \delta_y \end{pmatrix} \begin{pmatrix} \phi_{jx} \\ \phi_{jy} \end{pmatrix} = \begin{pmatrix} \Phi_j \\ \Psi_j \end{pmatrix} \tag{9}$$

from which the individual ϕ_{jx} and ϕ_{jy} can be obtained.

So all unknown parameters in (4) can be retrieved at the expense of $2n$ evaluations f_k and n evaluations F_k, or a mere total of $3n$ samples. In practice, when dealing with noisy real-life data, the value of n is overestimated by $\eta > n$. Moreover, the minimal number of $3\eta = 2\eta + \eta$ required samples for an η-term model of the form (4) is often again overestimated by $N + \mathfrak{n}$ with $N \geq 2\eta$ and $\mathfrak{n} \geq \eta$. The square $n \times n$ generalized eigenvalue problem (6), the $2n \times n$ Vandermonde system (7) and the $n \times n$ Vandermonde system (8) then respectively take the sizes $(N-\eta) \times \eta, N \times \eta$ and $\mathfrak{n} \times \eta$ and are all solved in the least squares sense. Often the generalized eigenvalue step is combined with a rank reduction step of the Hankel matrices to size $(N-\eta) \times \tilde{\eta}$ where $n < \tilde{\eta} < \eta$. Then in the generalized eigenvalue problem (6), the Vandermonde systems (7) and (8), the number of columns η is everywhere replaced by $\tilde{\eta}$.

B. Allowing coalescent inner products

We now describe how to proceed when some of the values $\exp(\Phi_j), j = 1, \ldots, n$ collide and the exponential sum therefore shrinks to $t < n$ terms. Such collisions or near-collisions happen easily since Φ_j is the projection of the vector ϕ_j on the vector Δ. Without loss of generality, we assume that the colliding terms are successive:

$$\begin{aligned} &\exp(\Phi_1) = \ldots = \exp(\Phi_{n_1}), \\ &\exp(\Phi_{n_1+1}) = \ldots = \exp(\Phi_{n_2}), \\ &\vdots \\ &\exp(\Phi_{n_{t-1}+1}) = \ldots = \exp(\Phi_n), \end{aligned}$$

with $n_0 = 0$ and $n_t = n$. The samples f_k then equal

$$f_k = \sum_{j=1}^{t} \left(\sum_{h=n_{j-1}+1}^{n_j} \alpha_h \right) \exp(\Phi_{n_j})$$

and the unknown coefficients in the now $t \times t$ linear systems (7) and (8) become respectively

$$\begin{aligned} \alpha_j, 1 \leq j \leq n \rightarrow \sum_{h=n_{j-1}+1}^{n_j} \alpha_h, 1 \leq j \leq t, \\ A_j, 1 \leq j \leq n \rightarrow \sum_{h=n_{j-1}+1}^{n_j} \alpha_h \exp(\Psi_h), 1 \leq j \leq t. \end{aligned}$$

To disentangle these coefficients and retrieve the individual $\alpha_1, \ldots, \alpha_n$ and $\Psi_1, \ldots, \Psi_n$ we need to repeat the shift procedure. Let us therefore collect samples at the shifted locations $k\Delta + \ell\delta$ for more values of ℓ:

$$F_{k\ell} = f(k\Delta_x + \ell\delta_x, k\Delta_y + \ell\delta_y), \qquad \ell = 1, \ldots, s-1, \quad k = 0, \ldots, t-1 \quad (10)$$

with $s \geq 2\max_j(n_j - n_{j-1})$ and $F_{k1} = F_k$. Since neither t nor s are actually known, their value is in practice overestimated again. We introduce the notation $A^{(\ell)} = (A_1^{(\ell)}, \ldots, A_t^{(\ell)})$ with

$$\begin{aligned} A_j^{(0)} &:= \sum_{h=n_{j-1}+1}^{n_j} \alpha_h, \\ A_j^{(\ell)} &:= \sum_{h=n_{j-1}+1}^{n_j} \alpha_h \exp(\ell\Psi_h), \qquad \ell = 1, \ldots, s-1, \end{aligned} \quad (11)$$

and replace (8) by the $s-1$ linear systems of size $t \times t$,

$$\begin{pmatrix} 1 & \cdots & 1 \\ \exp(\Phi_1) & \cdots & \exp(\Phi_t) \\ \vdots & & \vdots \\ \exp^{t-1}(\Phi_1) & \cdots & \exp^{t-1}(\Phi_t) \end{pmatrix} A^{(\ell)} = \begin{pmatrix} F_{0,\ell} \\ \vdots \\ F_{t-1,\ell} \end{pmatrix}, \qquad \ell = 1, \ldots, s-1. \quad (12)$$

For fixed j, the solutions $A_j^{(\ell)}, \ell = 0, \ldots, s-1$, where $A_j^{(0)}$ is computed from (7) with its size reduced to $t \times t$ and the $A_j^{(\ell)}, \ell = 1, \ldots, s-1$ are obtained from (12), provide values for an exponential analysis problem in its own right, as can be seen from (11). In the least squares version which is used when dealing with noisy data and t is overestimated, the systems (7) and (12) respectively take the size $N \times \eta$ and $\mathfrak{n} \times \eta$. Applying the basic technique from Section III-A reveals all unknowns and delivers the right hand sides for the 2×2 systems (9) where we let j run from 1 to n.

Since the vectors Δ and δ are linearly independent, it is impossible for distinct j and k to have $\Phi_j = \Phi_k$ and $\Psi_j = \Psi_k$ without having $\phi_j = \phi_k$ which are assumed distinct in (4). So we are sure that all terms are disentangled after performing and analyzing the $s-1$ shifts in (10).

C. Numerical computation

In what follows, we deal with 512×512 textures, which we extract from the central part of the image (if the input image is smaller then it is enlarged to 512×512 by interpolation). We choose the following values for the parameters: $\Delta = (1, 0), \delta = (0, 1), s = 512, t = 33, \eta = 170, N = 512, \mathfrak{n} = 512$. This choice is merely inspired by the size of the image and the consideration that $\eta \approx N/3$ works well for (6). As mentioned, we use a combination of the matrix pencil method studied in [10] with the rank reduction step described in [11] for the solution of (6), which constitutes the first step of the algorithm. We call this method the TLS-Prony method (as in [12]), because the first numerical method to perform exponential analysis was published by the French nobleman de Prony in 1795 [25]. In the second step of the algorithm, we analyze the Prony problem given in (11) using the decimated Prony-type algorithm presented in [23], with a decimation factor equal to 4. The use of decimation allows to build in some validation of the result, which is useful as the $\exp(\Psi_j)$ are in general less accurate than the $\exp(\Phi_j)$. The reason for this is that the various $\exp^\ell(\Psi_j)$ are obtained from solutions of the rather sensitive Vandermonde systems (12) and (8). The chosen decimation factor also reduces the system size of (6) from $(N - \eta) \times \eta$ to 75×53.

Similar to the thresholding that is performed on the DFT coefficients, we clean up the set of parameters computed so far. First of all, we project both $\exp(\Phi_j)$ and $\exp(\Psi_j)$ on the unit circle. Second, we discard the terms with small amplitude, meaning either with $|\alpha_j| < 0.002$ (for the images termed *chequered*) or with $|\alpha_j|$ less than $\max_j |\alpha_j|/160$ (for the images termed *woven*). After discarding these terms the coefficients α_j of the surviving terms are recomputed from the allover interpolation conditions

$$\sum_{j=1}^{n} \alpha_j \exp(\phi_{jx}k + \phi_{jy}\ell) = f(k, \ell), \qquad k, \ell = 0, \ldots, 511. \quad (13)$$

Finally, the resulting small amplitudes are discarded once more and the final α_j are recomputed from (13). At this moment the true value of n is revealed.

The thus resulting $\phi_j = (\phi_{jx}, \phi_{jy})$ computed from (9) can for instance be found in the second rows of Fig. 2 (synthetic image analysis) and Fig. 3 (real image analysis), where they are indicated by means of a red bullet. A comparison of the exponential frequency analysis output to the DFT spectrum, which is shown in both figures using green bullets, clearly confirms that the representation (4) is much sparser, while retaining a similar expressiveness. We point out that even for the DFT coefficients, in each image those coefficients that are in magnitude below $10^{-2.5}$ of the largest DFT coefficient are discarded to reduce the model complexity in the frequency space. In Fig. 2 where we deal with noisefree synthetic images (see appendix A for the parameters α_j, ϕ_j, n), the original image content is entirely retrieved by our method, as confirmed in Table I. In the respective rows 3 and 4 of both Fig. 2 and Fig. 3, image reconstructions based on both DFT and EA spectra are shown. The RMSE values and cardinality of the models used for the reconstructed images are again listed in Table I.

IV. Applications in computer vision

In Section III we derived the compact texture representation using exponential analysis and showed that original texture patterns can be reconstructed from a limited number of available samples. Those representations are valuable in many computer vision applications where the extraction of meaningful features is important. In this work we develop new algorithms to demonstrate two exemplar computer vision applications: 1) fabric defect detection and 2) texture classification.

A. Fabric defect detection

Generally, defect detection is an important but challenging task in computer vision applications [26] including industrial material inspection and fabric production [27]. We are

Algorithm 1: Fabric defect detection using EA.

Input: I: Fabric pattern (gray scale image)
Output: D: Defect detection mask (binary image)

for $i \leftarrow I_0$ **to** I_N, **do**
  $S \leftarrow exponential(I)$; // Sparse exponential terms.
  $R \leftarrow reconstruction(S)$; // Texture image reconstruction.
  $Q \leftarrow multissim(R, I)$; // Quality map of multi-scale SSIM.
  $Q_{bw} \leftarrow binarize(resize(Q\{3\}))$; // Resizing and binarization of mid-scale Q map.
  $M_E \leftarrow erode(Q_{bw}, se1)$; // Erosion with structural element $se1$ (radius 1).
  $M_D \leftarrow dilate(M_E, se2)$; // Dilation with structural element $se2$ (radius 10).
  $M_C \leftarrow clearborder(M_D)$; // Suppresses lighter structures which are connected to the border.
  $D \leftarrow remove\_SO(M_C, P)$; // Remove small objects (SO) with less pixels $P = 50$.
end
$E \leftarrow evaluation\_metric(D, I)$; // Calculate statistics such as detection success rate, true positive rates (TPR) etc.

Algorithm 2: Texture classification using EA.

Input: I, A: Texture image, actual class.
Output: P: Predicted class.

$T_S \leftarrow 150$; // T_S = number of exponential terms for feature vector formation.
for $i \leftarrow I_0$ **to** I_N, **do**
  $pre\_processing()$; // Resizing to 512×512 and gray scale conversion.
  $S \leftarrow exponential(I)$; // Sparse exponential terms.
  $S_R \leftarrow rank\_order(S)$; // Rank order S based on descending amplitudes.
  if $length(S) < T_S$ **then**
    $S_R \leftarrow S_R + zero\_padding$;
  else
    $S_R \leftarrow S_R(1 : T_S)$;
  end
  $F_{TR} \leftarrow concatenate(S_R)$; ; // Size of feature vector $F_{TR} = 70 * 5 = 350$.
end
$mdl = KNN(F_{TR}, A, K = 5)$; // Train KNN classifier.
$F_{TS} \leftarrow Repeat(line\ 3 - 13)$; // Create feature vector for test.
$P \leftarrow predict(F_{TS})$; // Predict classes for test images.
$E \leftarrow evaluation\_metric(A, P)$; // Calculate statistics, *e.g.*, confusion matrix.

particularly interested in fabric defect detection due to its wider applicability in the textile industry. In fact the demand of vision based automated fabric inspection attracts many researchers and practitioners in the manufacturing quality control automation domain [28]–[31].

State-of-the-art algorithms often rely on machine learning including deep learning based approaches that require large training data sets. As an alternative approach, we propose exponential analysis to represent the texture with limited model complexity, then reconstruct and compare with the input image in order to identify the defect within the pattern. The different steps constituting the proposed algorithm are shown in Algorithm 1. In identifying the differences between the input and the reconstructed images, firstly, we apply the Multi-Scale Structural Similarity Index Measure (MS-SSIM) through which a quality map is generated at different scales [32]. We use the mid-scale quality map to produce a difference image which in turn allows to detect and localise any defect(s) in the input texture pattern. We employ a couple of post-processing operations to create a clean binary defect mask. The post processing operations mainly include morphological operations such as erosion and dilation. The detected binary defect mask is then compared to the ground truth to produce evaluation statistics such as detection success rate, true positive rates etc. Details of these evaluation metrics, benchmark datasets and the results are described in Section V-C. It is worth noting that our approach does not require any machine learning steps, training data and hence will be of use to many similar applications.

B. Texture classification

We also explore the usage of our new decomposition tool in texture classification. Texture analysis is fundamental to most vision applications. Texture representation and classification are essential parts of the processing pipeline and are researched widely [20], [33], [34]. Our argument is that using the proposed exponential image analysis, texture can be represented compactly and used in feature engineering.

We build an imaging pipeline to extract the features. The steps of this pipeline are shown in Algorithm 2. Firstly, the input images are pre-processed which includes resizing and gray-scaling, followed by the estimation of a sparse exponential model as discussed in Section III. Five feature points, constituting of the amplitude modulus $|\alpha_j|$ and both real and imaginary part of the expressions $\exp(\langle \phi_j, \Delta \rangle)$ and $\exp(\langle \phi_j, \delta \rangle)$, are used to represent each term in (4). We concatenate these terms to generate the feature vector which is the input to a machine learning classifier such as K-nearest neighbour (KNN) which is used in this demonstrator with $K = 5$. As the estimated sparsity n of the exponential model relies on the input texture pattern, the method produces variable length feature vectors which are unsuitable for machine learning classifiers. To address this, we empirically set the number of exponential terms to be used in the feature vector. Either zero padding is applied whenever there are less terms

	DFT		Ours		Gain
	# Terms	RMSE	# Terms	RMSE	(%)
Synthetic images					
IM#1	111	0.0169	11	0.0012	90%
IM#2	231	0.0258	11	0.0015	95%
IM#3	279	0.0285	11	0.0011	96%
IM#4	163	0.0215	11	0.0011	93%
IM#5	279	0.0235	11	0.0011	96%
Real images from Oxford DTD dataset, category: woven					
001	1835	0.0469	167	0.0743	91%
003	793	0.0756	187	0.0780	76%
028	867	0.0562	163	0.0700	81%
038	1619	0.0723	159	0.1036	90%
064	4557	0.1100	107	0.1712	98%

TABLE I: Comparison of image decomposition using DFT and the proposed exponential analysis on synthetic and real images with respect to the number of terms to represent the texture, RMSE when reconstructed and % gain in model reduction.

or the model is truncated keeping the most significant terms in case there are more. We evaluate the performance on a benchmark data set and describe the results in Section V-D.

V. Experimental results and discussions[1]

To validate the proposed exponential analysis we perform experiments on both synthetic and real texture images and compare against DFT results, thus demonstrating the use and advantages of sparsity in the frequency domain. We also apply our method in common vision applications that require texture analysis, *e.g.*, fabric defect detection and texture classification. Within the scope of this work, our model considers only homogeneous patterns across the image and therefore does not work effectively on textures with multiple patterns.

A. Synthetic images

Following model (4), we generate five synthetic textures each containing 11 terms, the parameters of which are shown in Appendix A. We compare the exponential analysis results against DFT. For both, these results are used to reconstruct the images and compute root mean square error (RMSE) values. The results are shown in Fig. 2 and Table I. The proposed exponential analysis does not need prior knowledge on the number of terms and the validation step can threshold the unrelated terms automatically, which provides a reliable estimate of the total number of terms. As illustrated in Table I, the proposed method correctly retrieves the 11 exponential terms in the synthetic image, which significantly reduces the model complexity (more than 90% coefficient reduction against DFT) and guarantees a near perfect reconstruction (low RMSE).

B. Real texture images from benchmark dataset

Encouraged by the nice results for the synthetic images, we also conduct experiments with samples from the standard Oxford Describable Textures Dataset (DTD, category: woven) [35]. Similar to Section V-A, we compare the proposed methodology against the thresholded DFT in terms of number of coefficients and RMSE of the reconstructed image. The results are shown in Fig. 3 and Table I. The proposed method outperforms the DFT representation with fewer terms while retaining the RMSE values. However, our method does not work well in *woven_0064* or other images where the texture information is not sparse or consists of multiple patterns. This is due to the basic assumption of our model on sparsity and homogeneity, meaning exponential behaviour as in (4).

C. Fabric defect detection

In order to evaluate the new fabric defect detection algorithm, we test it on a publicly available benchmark dataset and compare with a recent Elo rating (ER) based method [28]. The dataset was developed by the Industrial Automation Research Laboratory, Dept. of Electrical and Electronic Engineering, The University of Hong Kong [36]. In this work we have used defected fabric texture images for two patterns: *box* and *star*. Both of them have five defect categories: *Thin Bar*, *Thick Bar*, *Netting Multiples*, *Broken End* and *Hole*. There are five samples in each category except *Thick Bar* in *Box* pattern which has six. In total we have experimented with 26 fabric samples from *Box* patterns and 25 samples from *Star* patterns.

In line with the reference ER method, five objective metrics are used in this work including *detection success rate* (DSR), *true positive rate* (TPR), *false positive rate* (FPR), *positive predictive value* (PPV), and *negative predictive value* (NPV) using the measurements such as *true positive* (TP), *false positive* (FP), *true negative* (TN) and *false negative* (FN). Pixel-wise comparisons are made between the Ground Truth (GT) and the detection output from our method. Objective metrics are calculated as

$$\begin{aligned} \mathrm{DSR} &= (\mathrm{TP}+\mathrm{TN})/(\mathrm{TP}+\mathrm{FN}+\mathrm{FP}+\mathrm{TN}),\\ \mathrm{TPR} &= \mathrm{TP}/(\mathrm{TP}+\mathrm{FN}),\\ \mathrm{FPR} &= \mathrm{FP}/(\mathrm{FP}+\mathrm{TN}),\\ \mathrm{PPV} &= \mathrm{TP}/(\mathrm{TP}+\mathrm{FP}),\\ \mathrm{NPV} &= \mathrm{TN}/(\mathrm{TN}+\mathrm{FN}). \end{aligned}$$

We have obtained two sets of results: a) visual results are shown in Fig. 4 and b) objective metrics with comparison are presented in Table II. We have shown samples from each category (*row 1*) for both patterns, the ground truth (*row 2*) and our detection results (*row 3*). Our method works well across the board and exhibits a closer match with the ground truth. Objective measures (represented as %) indicate acceptable performance for all categories and generally better performance when compared to the ER method [28]. Overall performances which are the averages of all categories show improvement over ER. Considering that our method does not depend on any training data, it has substantial potential where machine learning based approaches are not appropriate due to the lack of available samples. We like to stress that, the proposed fabric defect detection algorithm is only a demonstrator within the scope of this paper and can be further optimised by tuning the parameters and additional pre- and post-processing.

[1]Code will be made available as open source code for reproducibility.

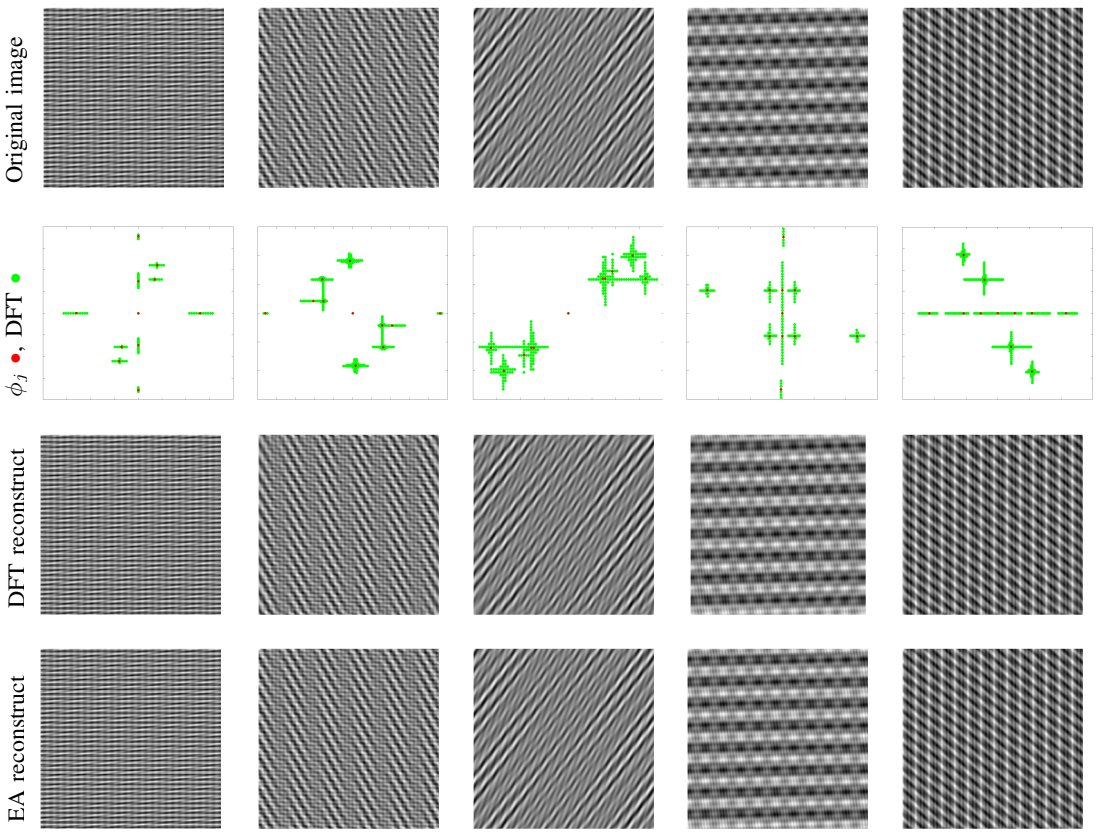


Fig. 2: Exponential analysis and comparison with DFT for synthetic noise-free images. Rows 1-4: original, exponential frequency analysis (ϕ_j in • overlaid on DFT frequency spectrum in •), reconstruction from thresholded DFT and reconstruction from exponential terms (EA) following (4).

D. Texture classification

We evaluate the texture classification on the publicly available Oxford Describable Textures Dataset (DTD) [35] that provides real world example images. As this demonstrator is a proof of concept we only choose a subset: categories *chequered* and *woven* which are relatively more aligned with the exponential analysis expectations, *i.e.*, the underlying texture pattern is homogeneous and periodic. All 160 images (80 in each category) are used in the experiment, with a random $70\% - 30\%$ $(56 - 24)$ training and testing split. Other parameters such as the number of exponential terms to form the feature vector ($T_S = 70$) or KNN classifier parameters ($K = 5$) are in line with the description in Algorithm 2. As each EA term constitutes of five feature points, the total feature vector length is $70 \times 5 = 350$. For evaluation purposes we generate the confusion matrix and overall accuracy.

Two sets of results are produced: a) visual results are shown in Fig. 5 and b) the confusion matrix is shown in Fig. 6. Visual results show examples of textures from the dataset used for the machine learning training, as well as correct and incorrect classifications. The confusion matrix indicates a higher classification accuracy for *chequered* (83.3%) compared to *woven* (66.7%) with an overall accuracy of 75.0%. Given the complexity of the real image textures, the proposed method works reasonably well, especially when the texture has a regular periodic pattern as in the *chequered* category.

To benchmark the performance of our approach, we also present the confusion matrix for the classification task using typical Speeded Up Robust Features (SURF) [37]. In this case we use the 100 strongest SURF feature points with an overall feature length of $100 \times 64 = 6400$ and preserve all other experimental set up, *i.e.*, same training and test data set, same KNN classifier with $K = 5$. Results are shown in Fig. 6 with an overall accuracy of 70.8%. In case of SURF, *woven* category works well due to the availability of stronger feature points.

The proposed exponential analysis based texture classification method achieves acceptable/better performance even with a significantly lower number of features (approximately 18 times less) when compared to SURF. This demonstrates the potential for a wider adaptation of exponential analysis in computer vision tasks. In its current form, the proposed approach is limited to regular texture patterns and therefore cannot process well any heterogeneous images. Future work includes the extension of the current approach to accommodate a wider set of images.

VI. Conclusions

This paper proposes a new image decomposition method using 2-d exponential analysis. We exploit key properties of exponential analysis, such as sparsity in the data fitting problem and continuity in the frequency space. The proposed

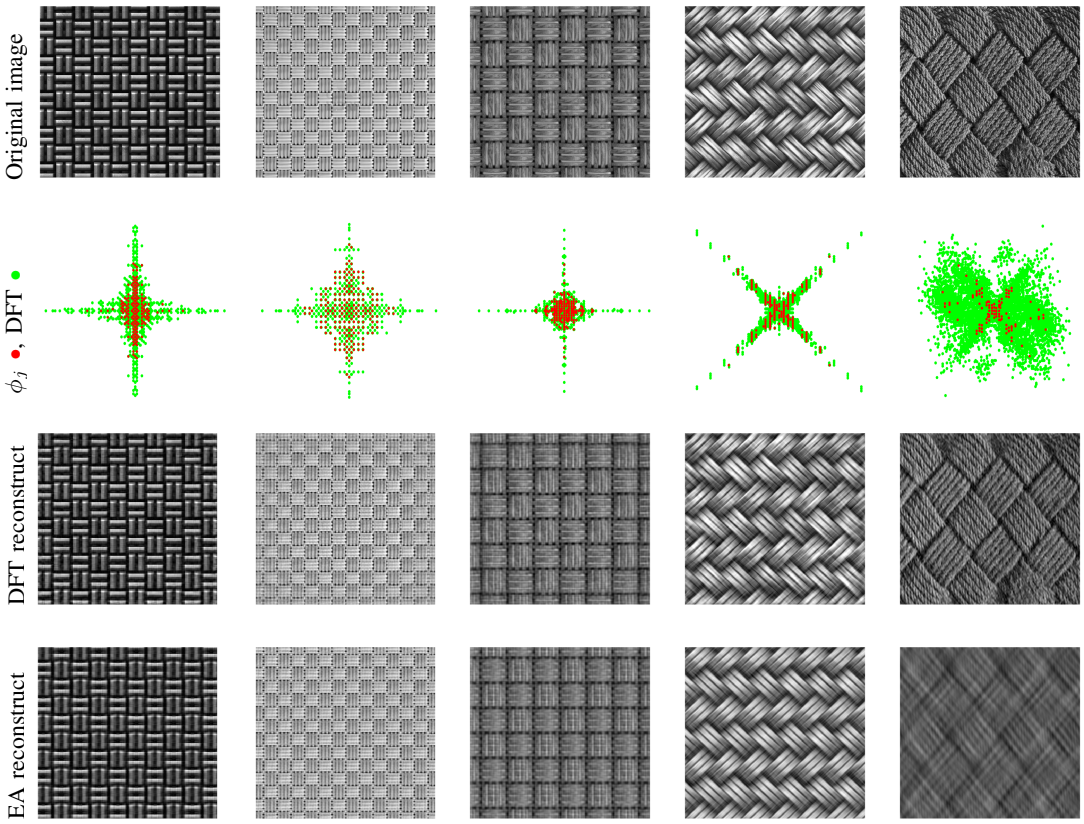


Fig. 3: Exponential analysis and comparison with DFT for real-life images. Rows 1-4: original, exponential frequency analysis (ϕ_j in • overlaid on DFT frequency spectrum in •), reconstruction from thresholded DFT and reconstruction from exponential terms following (4).

analysis results in a compact image representation in the frequency domain with significantly fewer coefficients. Experimental results outperform the classical DFT, ER in defect detection and SURF in texture classification, for both synthetic and real texture images and indicate new opportunities in the image processing domain.

The usefulness of our method is illustrated with two common vision applications, defect detection and texture classification, which are central components of many image processing algorithms. In particular, we propose a new algorithm for fabric defect detection which shows very acceptable performance on a benchmark dataset. Representing images with fewer terms will be beneficial for feature engineering in vision and can lead to better accuracy and a more efficient computation. We anticipate that the proposed EA based method can be part of a larger image processing pipeline and can benefit from pre- and post-processing optimisations.

REFERENCES

[1] Jae S. Lim, *Two-Dimensional Signal and Image Processing*, Prentice-Hall, Inc., USA, 1990.

[2] Gilbert Strang, "The discrete cosine transform," *SIAM Rev.*, vol. 41, no. 1, pp. 135–147, 1999.

[3] Stephane G Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," in *Fundamental Papers in Wavelet Theory*, pp. 494–513. Princeton University Press, 2009.

[4] Tianhorng Chang and C-CJ Kuo, "Texture analysis and classification with tree-structured wavelet transform," *IEEE Trans. Image Process.*, vol. 2, no. 4, pp. 429–441, 1993.

[5] Jean Claude Nunes, Yasmina Bouaoune, Eric Delechelle, Oumar Niang, and Ph Bunel, "Image analysis by bidimensional empirical mode decomposition," *Image and vision computing*, vol. 21, no. 12, pp. 1019–1026, 2003.

[6] Anna Linderhed, "Image empirical mode decomposition: A new tool for image processing," *Adv. Adapt. Data Anal.*, vol. 1, no. 02, pp. 265–294, 2009.

[7] Jerome Gilles, "Empirical wavelet transform," *IEEE Trans. Signal Process*, vol. 61, no. 16, pp. 3999–4010, 2013.

[8] Shishir Maheshwari, Ram Bilas Pachori, and U Rajendra Acharya, "Automated diagnosis of glaucoma using empirical wavelet transform and correntropy features extracted from fundus images," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 3, pp. 803–813, 2016.

[9] Annie Cuyt and Wen-shin Lee, "Multivariate exponential analysis from the minimal number of samples," *Adv. Comput. Math.*, vol. 44, pp. 987–1002, 2018, (Published online November 16, 2017).

[10] Yingbo Hua and Tapan K. Sarkar, "Matrix pencil method for estimating parameters of exponentially damped/undamped sinusoids in noise," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 38, pp. 814–824, 1990.

[11] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 7, pp. 984–995, July 1989.

[12] William M. Steedly, Ching-Hui J. Ying, and Randolph L. Moses, "Statistical analysis of TLS-based Prony techniques," *Automatica J. IFAC*, vol. 30, no. 1, pp. 115–129, 1994.

[13] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propag.*, vol. 34, no. 3, pp. 276–280, 1986.

[14] Yingbo Hua, "Estimating two-dimensional frequencies by matrix enhancement and matrix pencil," *IEEE Trans. Signal Process.*, vol. 40, no. 9, pp. 2267–2280, 1992.

[15] Stéphanie Rouguette and Mohamed Najim, "Estimation of frequencies and damping factors by two-dimensional ESPRIT type methods," *IEEE Trans. Signal Process.*, vol. 49, no. 1, pp. 237–245, 2001.

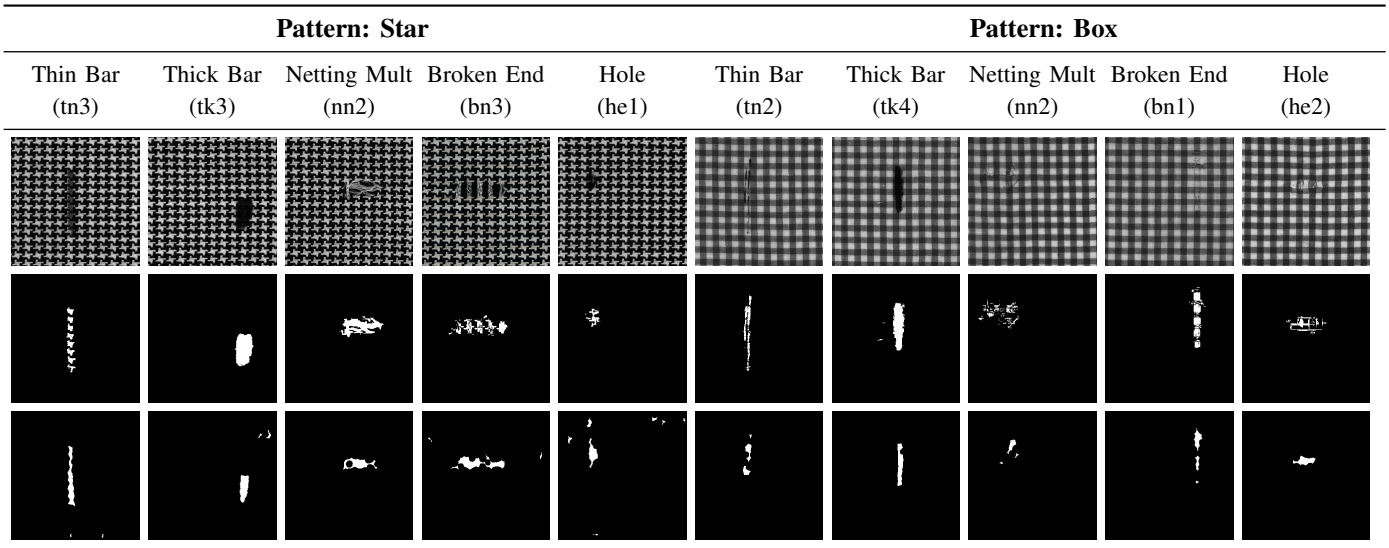


Fig. 4: Visual output from the proposed defect detection algorithm on benchmark data set. *Row 1*, *2* and *3* represent samples from each category, ground truth and detection output, respectively. Sample labels are given in () under column headings.

		Pattern: Star					Pattern: Box				
		DSR(%)	TPR(%)	FPR(%)	PPV(%)	NPV(%)	DSR(%)	TPR(%)	FPR(%)	PPV(%)	NPV(%)
Thin Bar (5)	ER [28]	96.72	45.47	2.83	12.50	99.45	93.41	5.84	4.51	2.36	97.68
	Proposed	99.26	41.14	0.26	57.38	99.51	99.24	26.41	0.08	82.63	99.31
Thick Bar (5)*	ER [28]	97.30	69.52	1.67	54.52	98.81	95.43	22.76	1.68	42.40	96.93
	Proposed	97.82	38.71	0.30	67.16	98.05	98.14	38.68	0.28	72.39	98.38
Netting Multiple (5)	ER [28]	97.76	16.42	0.82	12.61	98.54	95.77	0.15	0.04	4.00	95.81
	Proposed	98.76	28.64	0.14	66.20	98.90	98.71	17.11	0.30	43.05	99.00
Broken End (5)	ER [28]	98.13	8.79	1.16	7.17	99.27	95.27	10.26	0.69	30.43	95.88
	Proposed	99.09	45.06	0.47	36.63	99.55	98.52	32.96	0.17	80.87	98.67
Hole (5)	ER [28]	98.32	24.47	1.23	11.68	99.54	97.66	0	0.03	0	97.69
	Proposed	99.41	32.99	0.24	55.77	99.64	99.30	30.60	0.19	58.36	99.50
Overall	ER [28]	98.82	32.93	1.54	19.70	99.12	95.51	7.80	1.39	15.84	96.80
	Proposed	**98.87**	**37.31**	**0.28**	**56.63**	**99.13**	**98.78**	**29.15**	**0.20**	**67.46**	**98.97**

TABLE II: Defect detection performance and comparison with the state-of-the-art Elo rating (ER) [28] method. Numbers in brackets represent the number of samples in that category. *Thick Bar category has six images for the Box pattern.

[16] Swagata Nandi, Debasis Kundu, and Rajesh Kumar Srivastava, "Noise space decomposition method for two-dimensional sinusoidal model," *Comput. Statist. Data Anal.*, vol. 58, pp. 147–161, 2013.

[17] Thomas Peter, Gerlind Plonka, and Robert Schaback, "Prony's method for multivariate signals," *Proc. Appl. Math. Mech.*, vol. 15, pp. 665–666, 2015.

[18] Stefan Kunis, Thomas Peter, Tim Römer, and Ulrich von der Ohe, "A multivariate generalization of Prony's method," *Linear Algebra Appl.*, vol. 490, pp. 31–47, 2016.

[19] Tomas Sauer, "Prony's method in several variables," *Numerische Mathematik*, vol. 136, pp. 411–438, 2017.

[20] Li Liu, Jie Chen, Paul Fieguth, Guoying Zhao, Rama Chellappa, and Matti Pietikäinen, "From BoW to CNN: Two decades of texture representation for texture classification," *Int. J. Comput. Vis.*, vol. 127, no. 1, pp. 74–109, 2019.

[21] Yuan Hou, Annie Cuyt, Wen-shin Lee, and Deepayan Bhowmik, "Decomposing textures using exponential analysis," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 1920–1924.

[22] Annie Cuyt and Wen-shin Lee, "How to get high resolution results from sparse and coarsely sampled data," *Appl. Comput. Harmon. Anal.*, vol. 48, pp. 1066–1087, 2020, (Published online October 11, 2018).

[23] M. Briani, A. Cuyt, F. Knaepkens, and W.-s. Lee, "VEXPA: Validated EXPonential Analysis through regular subsampling," *Signal Processing*, 2020, (Published online July 17, 2020).

[24] Annie Cuyt and Wen-shin Lee, "Sparse interpolation and rational approximation," Providence, RI, 2016, vol. 661 of *Contemporary Mathematics*, pp. 229–242, American Mathematical Society.

[25] R. de Prony, "Essai expérimental et analytique sur les lois de la dilatabilité des fluides élastiques et sur celles de la force expansive de la vapeur de l'eau et de la vapeur de l'alkool, à différentes températures," *J. Ec. Poly.*, vol. 1, no. 22, pp. 24–76, 1795.

[26] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger, "MVTec AD–A comprehensive real-world dataset for unsupervised anomaly detection," in *Proc. IEEE CVPR*, 2019, pp. 9592–9600.

[27] Kazım Hanbay, Muhammed Fatih Talu, and Ömer Faruk Özgüven, "Fabric defect detection systems and methods—a systematic literature review," *Optik*, vol. 127, no. 24, pp. 11960–11973, 2016.

[28] Colin SC Tsang, Henry YT Ngan, and Grantham KH Pang, "Fabric inspection based on the Elo rating method," *Pattern Recognition*, vol. 51, pp. 378–394, 2016.

[29] Chunlei Li, Guangshuai Gao, Zhoufeng Liu, Miao Yu, and Di Huang, "Fabric defect detection based on biological vision modeling," *IEEE Access*, vol. 6, pp. 27659–27670, 2018.

[30] Boshan Shi, Jiuzhen Liang, Lan Di, Chen Chen, and Zhenjie Hou, "Fabric defect detection via low-rank decomposition with gradient

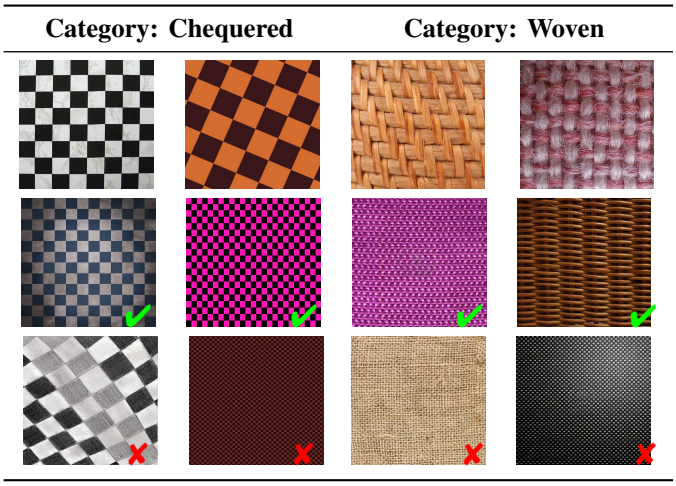


Fig. 5: Visual results for texture classification using exponential analysis. *Row 1:* training samples from the DTD dataset, *Row 2:* examples of correct classification, *Row 3:* examples of mis-classification.

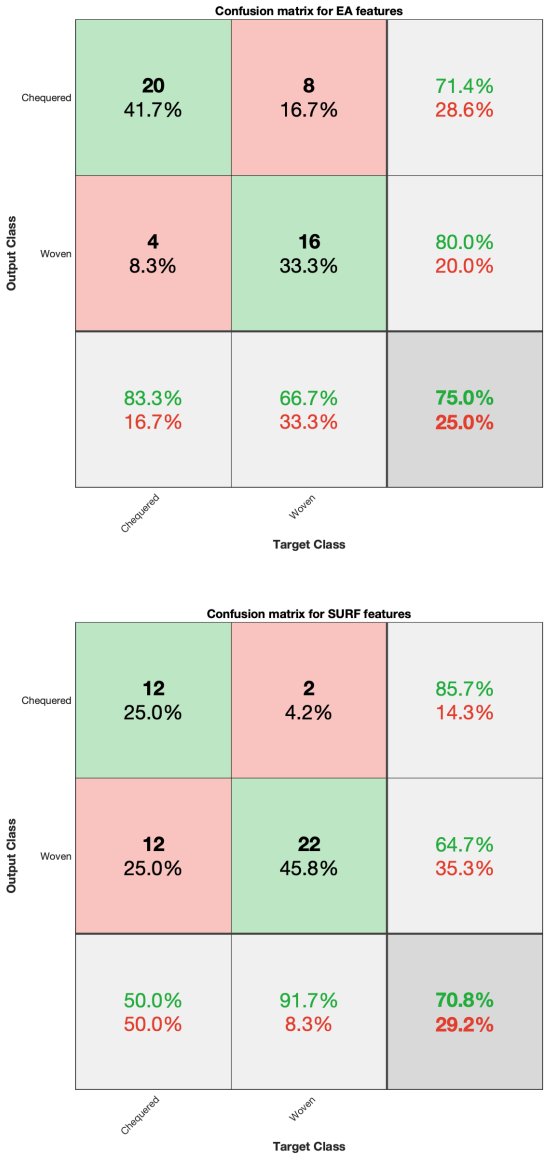


Fig. 6: Texture classification performance evaluation. Confusion matrix *row 1*: classification using proposed exponential analysis (feature vector length: **350**), and *row 2*: comparison with standard SURF features (feature vector length: **6400**). The proposed EA method achieves similar performance with much smaller feature length (gain ≈ 95%).

information and structured graph algorithm," *Information Sciences*, vol. 546, pp. 608–626, 2021.

[31] Juhua Liu, Chaoyue Wang, Hai Su, Bo Du, and Dacheng Tao, "Multistage GAN for fabric defect detection," *IEEE Trans. Image Process.*, vol. 29, pp. 3388–3400, 2019.

[32] Zhou Wang, Eero P Simoncelli, and Alan C Bovik, "Multiscale structural similarity for image quality assessment," in *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*. Ieee, 2003, vol. 2, pp. 1398–1402.

[33] Rakesh Mehta and Karen Egiazarian, "Texture classification using dense micro-block difference," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1604–1616, 2016.

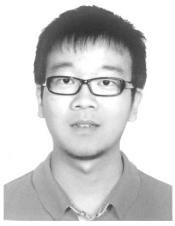
[34] Wei Zhang, Weidong Zhang, Kan Liu, and Jason Gu, "A feature descriptor based on local normalized difference for real-world texture classification," *IEEE Transactions on Multimedia*, vol. 20, no. 4, pp. 880–888, 2017.

[35] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi, "Describing textures in the wild," in *Proc. IEEE CVPR*, 2014, pp. 3606–3613.

[36] Henry Y.T. Ngan and Grantham K.H. Pang, "Fabric defect dataset," 2009, dataset was developed by Industrial Automation Research Laboratory, Dept. of Electrical and Electronic Engineering, The University of Hong Kong, and available at https://ytngan.wordpress.com/codes/ [Accessed on 07-July-2021].

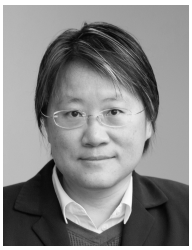
[37] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, "SURF: Speeded up robust features," in *Proc. European Conference on Computer Vision (ECCV)*. Springer, 2006, pp. 404–417.

Deepayan Bhowmik received the Ph.D. in Electronic and Electrical Engineering from the University of Sheffield, UK in 2011. Previously he worked as a lecturer in Computing at Sheffield Hallam University, a research associate at Heriot-Watt University, & the University of Sheffield and a systems engineer in ABB Ltd. Currently he is working as a lecturer at University of Stirling, UK. He received UK EPSRC Dorothy Hodgkin Postgraduate Award and multiple research council and industrial funding to conduct research in image and other multi-modal signal processing, embedded imaging and applications such as security, surveillance and remote sensing.



Yuan Hou obtained his BSc and MSc degrees in Software Engineering from East China Normal University in Shanghai, China, in 2014 and 2017 respectively. Since October 2017, he is a PhD student in the Computational Mathematics (CMA) research group at the University of Antwerp in Belgium. His research focuses on using state-of-art multivariate exponential analysis techniques to tackle challenges in identified computational science and engineering applications.

Annie Cuyt received the Doctor Scientiae degree (summa cum laude) from the University of Antwerp, Belgium, in 1982, with the felicitations of the jury. Subsequently, she was a Research Fellow of the Alexander von Humboldt Foundation, Germany. She obtained the Habilitation in 1986, and was honoured with a Masuda Research Grant, Japan. She is currently a Full Professor with the Department of Mathematics and Computer Science, University of Antwerp. Her current research interests are in numerical approximation theory and its numerous applications in computational science and engineering. In view of her expertise, she has served on many national and international science foundation boards. Annie Cuyt is a life-time member of the Royal Flemish Academy of Belgium for the Sciences and Arts since 2014.



Wen-shin Lee (Member, IEEE) received the Ph.D. degree in computational mathematics from North Carolina State University, USA, in 2001. After then, she was a Post-Doctoral Fellow with the University of Waterloo, Canada, and INRIA, France. In 2005, she moved to Belgium and became a member of the Computational Mathematics Research Group, University of Antwerp. She is Lecturer with the Division of Computing Science and Mathematics, University of Stirling, U.K. Her research background is computer algebra and symbolic-numeric computation. Her current focus is on exponential analysis, especially its connection to sparse interpolation and applications in signal processing. She is the Editor of the ACM Communications in Computer Algebra.

APPENDIX A
PARAMETERS FOR SYNTHETIC IMAGES

IM#1			IM#2			IM#3			IM#4			IM#5		
$\phi_{jx}/2\pi\mathrm{i}$	$\phi_{jy}/2\pi\mathrm{i}$	α_j	$\phi_{jx}/2\pi\mathrm{i}$	$\phi_{jy}/2\pi\mathrm{i}$	α_j	$\phi_{jx}/2\pi\mathrm{i}$	$\phi_{jy}/2\pi\mathrm{i}$	α_j	$\phi_{jx}/2\pi\mathrm{i}$	$\phi_{jy}/2\pi\mathrm{i}$	α_j	$\phi_{jx}/2\pi\mathrm{i}$	$\phi_{jy}/2\pi\mathrm{i}$	α_j
0	0	0.371	0	0	0.493	0	0	0.348	0	0	0.803	0	0	0.573
0.074	0	0.117	0.021	-0.031	0.446	-0.029	-0.037	0.071	-0.016	0	0.717	-0.031	0.021	0.305
-0.074	0	0.117	-0.021	0.031	0.446	0.029	0.037	0.071	0.016	0	0.717	0.031	-0.021	0.305
-0.178	0	0.033	-0.0215	0.0415	0.428	-0.024	-0.031	0.064	-0.053	-0.002	0.272	0	-0.027	0.247
0.178	0	0.033	0.0215	-0.0415	0.428	0.024	0.031	0.064	0.053	0.002	0.272	0	0.027	0.247
0.111	0.016	0.029	-0.091	0.003	0.31	0.024	0.029	0.05	-0.016	-0.02	0.243	0	0.054	0.125
-0.111	-0.016	0.029	0.091	-0.003	0.31	-0.024	-0.029	0.05	0.016	0.02	0.243	0	-0.054	0.125
-0.078	-0.014	0.028	-0.059	0.032	0.219	0.024	0.065	0.048	0.016	-0.02	0.24	0.054	-0.054	0.105
0.078	0.014	0.028	0.059	-0.032	0.219	-0.024	-0.065	0.048	-0.016	0.02	0.24	-0.054	0.054	0.105
0	-0.052	0.026	0	0.092	0.186	-0.04	-0.054	0.048	-0.016	0.118	0.223	0	-0.108	0.103
0	0.052	0.026	0	-0.092	0.186	0.04	0.054	0.048	0.016	-0.118	0.223	0	0.108	0.103