Invariant Image Classification Using Triple-Correlation-Based Neural Networks

Anastasios Delopoulos, *Member, IEEE*, Andreas Tirakis, and Stefanos Kollias, *Member, IEEE*

***Abstract*—Triple-correlation-based neural networks are introduced and used in this paper for invariant classification of two-dimensional gray scale images. Third-order correlations of an image are appropriately clustered, in spatial or spectral domain, to generate an equivalent image representation that is invariant with respect to translation, rotation, and dilation. An efficient implementation scheme is also proposed, which is robust to distortions, insensitive to additive noise, and classifies the original image using adequate neural network architectures applied directly to 2-D image representations. Third-order neural networks are shown to be a specific category of triple-correlation-based networks, applied either to binary or gray-scale images. A simulation study is given, which illustrates the theoretical developments, using synthetic and real image data.**

I. Introduction

INVARIANT pattern recognition is one of the hardest families of problems in the theory of perception and in computer vision. The problem of invariant recognition is to understand how our perception of an object remains unaffected in spite of the considerable changes that the retinal image of an object may undergo [7]. Among the most important of these changes are translation, scale, and rotation, as well as distortion and addition of noise [25].

The use of neural networks for invariant image recognition has been recently a topic of extensive research. Various types of approaches have been proposed in the literature for this purpose. The most common approach to the problem of invariant recognition consists of two steps; in the first a feature extraction or transformation method provides an invariant image representation, or a set of features of reduced dimensionality, while a neural network is used in the second step to classify the unknown images, based on the extracted features [17], [33]. Other techniques, such as the neocognitron [8], [9], or the labeled graph matching approach [1], [22], derived from biological concepts, attempt to implement physiological mechanisms in artificial neural systems. These techniques, which generally are of high complexity, are not considered in this paper.

The need for invariant recognition translates, therefore, to the requirement for extraction of features which are invariant with respect to transformations of the input image. The loss of information, caused by the transformation of the input image space to the feature or equivalent image representation space, as well as the robustness of image classification in the presence of small distortions or of additive noise, are the most crucial factors in the design of such a recognition scheme.

Multilayer perceptrons have been widely examined in the neural network field, as a tool for signal classification, based on the extraction of appropriate features from signals. Error-feedback supervised learning algorithms, such as backpropagation, are generally used to train a multilayer feed-forward network. A crucial aspect concerning the network performance is generalization, i.e., the ability of the network to classify correctly input data, which were not included in its training set. Good generalization is a result of appropriate network design; a small number of interconnection weights (i.e., free parameters during training) should be generally used for this purpose, and any *a priori* knowledge about the problem should be included in the network architecture. Consequently, structured networks of small size are likely to have a better generalization.

As far as invariant classification is concerned, use of multilayer perceptrons would require exhaustive training over the patterns and all possible transformations of them, since it has not been evident whether hidden units in first-order neural networks can efficiently represent invariant feature sets [29]. For this reason, it has been proposed to use higher-order networks [12], [13], which take advantage of information about expected relationships between the input nodes of the network [27]. An important category of higher-order networks are the third-order ones, because they can provide simultaneous invariance with respect to translation, scale, and rotation.

The ith unit in a third-order neural network, shown in Fig. 1, computes its output y_i, as follows

$$y_i = f\left(\sum_j \sum_k \sum_l w_{ijkl} x_j x_k x_l\right), \tag{1}$$

where x_i denotes the ith element of the input pattern, or equivalently the ith input unit, w_{ijkl} is the weight connecting the product of j, k, and l input units to the ith output unit, and f is the sigmoid function. The summations extend over the whole input space, so that the limits are defined by the problem.

The use of third-order networks for invariant recognition of binary images has been examined in [27], [28], [19]. Let us consider an object, represented in two dimensions and in binary form. Then, three points within it, say j, k, l, define a triangle with included angles (a,b,c), as shown in Fig. 2. Any translation, dilation and/or rotation of the object transforms

Manuscript received October 30, 1991; revised December 7, 1992. This work was partly supported by the the European ESPRIT 2092 project ANNIE (Artificial Neural Networks for Industry in Europe), 1989–1991.

The authors are with the Computer Science Division, National Technical University of Athens, Athens, Greece.

IEEE Log Number 9208465.

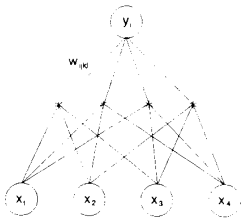


Fig. 1. Third-order neuron model.

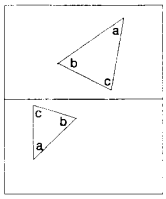


Fig. 2. An input triplet and scaled/rotated version of it.

(jkl) to another set of points $(j'k'l')$ preserving (a,b,c). If a third-order neural network is employed for invariant recognition of this object, then it should treat the inputs at points j', k', l' in the same way with those at j, k, l, i.e., the network weights w_{ijkl} associated with triplets of input points forming *similar* triangles should be equal to each other. This can be written as follows,

$$w_{ijkl} = w_{iabc} \equiv w_{icab} \equiv w_{ibca}, \tag{2}$$

where the weight is expressed as a function of the angles a,b,c. In general, though, $w_{iabc} \neq w_{ibac}$, unless recognition is reflection insensitive. Equation (2) yields constraints on the network interconnection weights and thus imply a particular structure of the network, providing it with the ability to perform invariant recognition of the object under consideration. However, the ability of such a network to provide solutions to complex real-life problems is still an open problem, mainly due to the excessive number of interconnections that is required in cases of a large input image size.

In this paper it is shown that image-recognition using third-order neural networks is a specific case of a more general image recognition scheme using triple-correlation-based neural networks.

Third-order signal correlations are higher-order signal statistics [23] with two important properties:

1) In contrast to second-order correlations, and apart from some special cases (including transient signals and 2-D co-sinusoids), in which third-order-correlations vanish, triple-correlations of *deterministic* signals have an one-to-one correspondence with the original signal (except of a shift ambiguity).
2) Third-order-correlations of zero-mean non-skewed noise (such as Gaussian or linear and symmetrically distributed) are zero in the mean and furthermore, they tend to zero w.p.1 as the size of the available data record tends to infinity.

The first property generally yields a complete description of the signal based on its triple-correlation. On the other hand, the second property can be used, under certain conditions, to improve the SNR in applications where the signal under consideration is corrupted by non-skewed additive noise. Based on their properties, third order correlations can be very advantageous for image recognition, if invariance is included in them. Various invariant classification techniques based on third-order-correlation and/or bispectrum, i.e., the frequency domain counterpart of third-order-correlation, of 2-D signals have been recently reported. In [30], [31], and [32], the triple-correlation of the signal is first computed and transformed to log-polar grid; then, the amplitude of the Fourier transform of the resulting 4-D signal is used as a shift, rotation, and dilation invariant feature. To overcome the high computational load of the derived algorithm, it is also proposed to use only the 0th "slice" of the 4-D triple correlation domain as feature, sacrifying the uniqueness of the representation. In [2] feature sets generated by integrating the bispectrum of 1-D signals along straight lines in the bifrequency plain are used for translation, dc level, and amplification invariant 1-D pattern classification. In [3] the above features, computed over a discrete set of projections of an image, are proposed for invariant 2-D object recognition. Uniqueness of the representation is claimed due to the one-to-one correspondence between the image and the collection of all projections of it (Radon transform). Triple-correlations and bispectra are also considered as features for the classification of random 2-D patterns (textures) in [10], [11], [31], and [32].

In the present paper we use an appropriate clustering of the 4-D triple-correlation domain of 2-D images, in order to obtain translation/dilation/rotation invariant representations and then propose appropriately structured neural network architectures to classify these representations. We describe a systematic procedure which gradually departs from uniqueness, reducing the size of the representations as well as the corresponding computational load. This reduction, which is equivalent to local averaging in the triple-correlation domain, is shown to generate a representation that is robust with respect to small object distortions and additive noise. Furthermore, we show that the previously mentioned third-order neural networks are a specific category of the proposed neural network architectures, when fed with a reduced version of the novel feature set.

The structure of the paper is as follows: Section II introduces the novel invariant object representation, after quoting some definitions and properties of the triple-correlation of an image. Section III describes an efficient implementation scheme for obtaining such a representation, that can be used in practical applications and simultaneously suppress the effect of non-skewed noise. Appropriate neural network architectures, applied directly to triple-correlation-based image representations, are presented in Section IV for invariant classification of the images. Simulation results are given in Section V which illustrate all theoretical developments, including comparisons with other classification techniques and application to synthetic and real image data.

II. Invariant Representation of Gray-Scale Images

A new representation of zero-mean gray-scale images is defined in this section, that has the following properties:

1) shift-rotation-scale invariance (SRS).
2) unique correspondence between the class of original images that are mutually related with rotation- translation-scaling transformation and the new representation domain, provided that their triple-correlation does not vanish.
3) noise insensitivity.

As mentioned in the Introduction, the use of triple correlation for SRS invariant recognition is also proposed in [32]. In the same reference a presentation of other invariant recognition schemes is made. What we propose here is a systematic extraction of a novel type of features/images that are functions of the triple correlation lags, in a way that allows:

1) Computationally efficient implementation of the classification using an artificial neural network.
2) Reduction in the size of the problem by *gradually* abolishing the uniqueness of the representation in favor of the computational simplicity.

Before proceeding to define the new representation, we quote some fundamental properties of the triple correlation of a 2-D signal.

A. Definition and Properties of Third-Order Correlations

Let $x(\boldsymbol{t})$ be a real 2-D signal with support $S = (0,T) \times (0,T)$. Its triple correlation is defined as,

$$x_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) \triangleq \frac{1}{T^2}\int_S x(\boldsymbol{t})x(\boldsymbol{t}+\boldsymbol{\tau}_1)x(\boldsymbol{t}+\boldsymbol{\tau}_2)d\boldsymbol{t}. \quad (3)$$

where $\boldsymbol{\tau}_1$, $\boldsymbol{\tau}_2$ are also 2-D vectors in $S' = [-T\cdots T] \times [-T\cdots T]$. In the discrete case, respectively, i.e., when $\boldsymbol{t} \in S = [0\cdots N-1]\times[0\cdots N-1]$,

$$x_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) \triangleq \frac{1}{N^2}\sum_S x(\boldsymbol{t})x(\boldsymbol{t}+\boldsymbol{\tau}_1)x(\boldsymbol{t}+\boldsymbol{\tau}_2). \quad (4)$$

with $\boldsymbol{\tau}_1$, $\boldsymbol{\tau}_2$ defined in $S' = [-(N-1).\cdots,(N-1)]\times[-(N-1),\cdots,(N-1)]$.

1) Symmetries: Triple-correlations of 2-D signals have the following symmetries:

$$\begin{aligned} x_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) &= x_3(\boldsymbol{\tau}_1-\boldsymbol{\tau}_2,-\boldsymbol{\tau}_2) = x_3(\boldsymbol{\tau}_2-\boldsymbol{\tau}_1,-\boldsymbol{\tau}_1)\\ &= x_3(\boldsymbol{\tau}_2,\boldsymbol{\tau}_1) = x_3(-\boldsymbol{\tau}_1,\boldsymbol{\tau}_2-\boldsymbol{\tau}_1)\\ &= x_3(-\boldsymbol{\tau}_2,\boldsymbol{\tau}_1-\boldsymbol{\tau}_2). \end{aligned} \quad (5)$$

2) Noise Insensitivity:

a) Let $e(\boldsymbol{t})$ be a zero-mean additive Gaussian or linear and symmetrically distributed noise.

Then,

$$E\{e_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)\} = 0, \quad (6)$$

and also under common mixing conditions,

$$e_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) \to 0 \text{ as } T \to \infty. \quad (7)$$

Moreover, if $x(\boldsymbol{t})$ is a deterministic object and $y(\boldsymbol{t}) = x(\boldsymbol{t}) + e(\boldsymbol{t})$, then

$$E\{y_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)\} = x_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)(\forall \boldsymbol{\tau}_1,\boldsymbol{\tau}_2. \quad (8)$$

Expectation in (8) is w.r.t. the random component $e(\boldsymbol{t})$. Under certain conditions, about the existence of the limit of $x_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ as $T \to \infty$, and provided that $e(\boldsymbol{t})$ is of "fast" decaying memory, then also $y_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) \to x_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ in the mean-square-sense (MSS) as $T \to \infty$ (see [5]). Unfortunately, the conditions in [5] are not generally met in situations of 2-D object recognition. Hence, noise insensitivity can be obtained using (8), only if, say L, independent noisy records are available, in which case the expectation in (8) is approximated (in view of the Law of Large Numbers) by,

$$\frac{1}{L}\sum_{i=1}^{L} y_3^{(i)}(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2). \quad (9)$$

b) Despite the result of part a), one can improve the SNR of the triple-correlation domain, performing appropriate averaging of the triple-correlation lags computed from a single data record. Indeed in [5] it has been proved that, under relaxed mixing conditions on $e(\boldsymbol{t})$, the lags of $y_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ are assympotically normal and in pairs independent; this assertion has been proved for 1-D signals and it is straightforward to extend it to 2-D signals as well. Consequently and provided that the object region of extent is sufficiently large, one can claim using Kinchin's theorem (see e.g., [26], p. 193) that,

$$\frac{1}{L}\sum_{(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)\in A} y_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) \to \frac{1}{L}\sum_{(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)\in A} x_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) \quad \text{in Probability as } L \to \infty, \quad (10)$$

where A is a set of lags and L is its cardinal number. In addition, according to [5], $var\{y_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)\} = O(1/T^2)$, if $L/T^2 \to 0$ as $T \to \infty$; thus, Tshebycheff's condition (see e.g., [26], p. 193) is satisfied and hence (10) holds in the mean-square-sense. The practical meaning of (10) is that averaging of triple-correlation lags suppresses the effects of non-skewed noise at the cost of losing some information contained in $x_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$. This result is used in Section III in order to show that the proposed invariant representation is also noise resistant.

3) One-to-One Relation: Provided that $x_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ is not identically zero, there is an one-to-one correspondence between a 2-D signal $x(\boldsymbol{t})$ and its triple correlation, i.e.,

$$x_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) = y_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) <=> x(\boldsymbol{t}) = y(\boldsymbol{t}). \quad (11)$$

This property implies that, in general, we can move indistinguishably from the signal domain to the triple correlation domain without loss of information or, in other words, we can distinguish two signals by comparing their triple correlations.

4) Shift-Rotation-and-Scaling Properties: Let $y(\boldsymbol{t}) = x(\boldsymbol{T}_{\alpha,\theta}\boldsymbol{t} + \boldsymbol{t}_0)$, where

$$\boldsymbol{T}_{\alpha,\theta} = \alpha\begin{bmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{bmatrix}. \quad (12)$$

is a scaling and rotation matrix and $\boldsymbol{t}_0$ a shifting vector. Provided that the application of these transforms is possible

within the limits of integration in (3) and (4), i.e., within the region of support S, it can be easily checked out that,

$$y_3(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2) = x_3(\boldsymbol{T}_{\alpha,\theta}\boldsymbol{\tau}_1, \boldsymbol{T}_{\alpha,\theta}\boldsymbol{\tau}_2). \tag{13}$$

The above statement implies that when the signal plane shifts, the triple correlation is unaffected, and when the signal plane rotates and/or is rescaled by $\boldsymbol{T}$, the same happens in the triple correlation domain for both lag indices $\boldsymbol{\tau}_1, \boldsymbol{\tau}_2$. A formal proof of (13) based on bifrequency domain arguments can be found in [32]. In the following, we shall consider the continuous case and only reaching the end of Section II we shall return to the discrete definition, which is convenient for implementation purposes.

B. The Proposed Representation

The triple correlation of a 2-D signal $x(\boldsymbol{t})$ is a function of two 2-D vector indices, $\boldsymbol{\tau}_1, \boldsymbol{\tau}_2$, each of them spanning the subset S' of $\boldsymbol{R}^2$. Let us start clustering the 4-D triple-correlation domain; by definition, $x_3(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ is the accumulation of all triple products formed by the values of $x(\boldsymbol{t})$ that lie on the corners of those *equal* triangles that are *shifts* of a prototype triangle defined by arbitrary vectors $\boldsymbol{\tau}_1, \boldsymbol{\tau}_2$. Hereafter we shall call $W(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ the set of all these triangles. Define, next, the set $K(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ of all triangles that are *similar* to the members of $W(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$, i.e.,

$$K(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2) = \bigcup_{\boldsymbol{T}_{\alpha,\theta}} W(\boldsymbol{T}_{\alpha,\theta}\boldsymbol{\tau}_1, \boldsymbol{T}_{\alpha,\theta}\boldsymbol{\tau}_2). \tag{14}$$

where

$$\boldsymbol{T}_{\alpha,\theta} = \alpha \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}, \alpha > 0, \theta \in [-\pi, \pi]. \tag{15}$$

For any set $K(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$, we define a corresponding *class* $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ of triple correlation lags; according to our notation, class $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ contains all lags that are associated with a subset $W(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ of $K(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$. In other words $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ is the set of all triple-correlation lags whose indices form, on the $\boldsymbol{R}^2$ plane, triangles similar to the triangle defined by the vectors $\boldsymbol{\tau}_1, \boldsymbol{\tau}_2$. In view of (2) and the related discussion, this clustering will be proved to be useful in the handling of rotations and dilations of the object. Two comments should be made at this point:

1) Let $\boldsymbol{\tau}_1, \boldsymbol{\tau}_2$ span the entire S'; then identical classes will be generated for different indices $(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$, $(\boldsymbol{\tau}_1', \boldsymbol{\tau}_2')$, if these indices form similar triangles, or, if they are related through symmetries of the triple correlation. This redundancy will be removed later, imposing certain restrictions on $\boldsymbol{\tau}_1, \boldsymbol{\tau}_2$ (see Proposition 4).
2) Since $\boldsymbol{\tau}_1, \boldsymbol{\tau}_2$ are allowed to take values on $\boldsymbol{R}^2$ (not necessarily on a discrete grid) there will be uncountable distinct classes $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ and moreover each of these classes will include an uncountable set of distinct lags. Notice that this happens even for finite signal support $T \times T$. In Section III, where the discrete implementation is discussed, appropriate quantization is introduced which yields representations of finite dimensions.

The main contribution of the following Propositions 1–3 is to state that $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ are appropriately defined such that rotation and/or dilation does not cause inter-class interference, while resulting in an internal circular shift of the content of each class.

Proposition 1: Any rotation θ and/or scaling α of the original 2-D plane, $x(\boldsymbol{t})$, results in an internal rearrangement of the elements of $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$, $(\forall \boldsymbol{\tau}_1, \boldsymbol{\tau}_2 \in S'$.

Proof: It is a consequence of (13) and the definition of $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$, since any such transformation will translate the specific $W(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ subset to another subset in $K(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$. It is interesting to notice that no inter-class interference is caused by any rotation, shift or scaling.

We next define the following arrangement between the members of each class $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$:

$$\tilde{x}_3(\rho, \phi; \boldsymbol{\tau}_1, \boldsymbol{\tau}_2) \triangleq x_3(\boldsymbol{T}_{\beta,\phi}\boldsymbol{\tau}_1, \boldsymbol{T}_{\beta,\phi}\boldsymbol{\tau}_2), \tag{16}$$

where,

$$\boldsymbol{T}_{\beta,\phi} = \beta \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix}, \text{and} \rho = \log\beta.$$

Variables ρ and ϕ are introduced to represent any scaled (in log form) and rotated triangle $W(\boldsymbol{T}_{\beta,\phi}\boldsymbol{\tau}_1, \boldsymbol{T}_{\beta,\phi}\boldsymbol{\tau}_2)$ when compared to a prototype triangle of class $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$.

Proposition 2: Let $y(\boldsymbol{t}) = x(\boldsymbol{T}_{\alpha,\theta}\boldsymbol{t} + \boldsymbol{t}_0)$ where $\boldsymbol{T}_{\alpha,\theta}$ is defined as in (12). Then, in the interior of each class $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$

$$\tilde{y}_3(\rho, \phi; \boldsymbol{\tau}_1, \boldsymbol{\tau}_2) = \tilde{x}_3(\rho + \log\alpha, \phi + \theta; \boldsymbol{\tau}_1, \boldsymbol{\tau}_2). \tag{17}$$

Proof: Applying (16), for each class $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$, we have that $\tilde{y}_3(\rho, \phi; \boldsymbol{\tau}_1, \boldsymbol{\tau}_2) = y_3(\boldsymbol{T}_{\beta,\phi}\boldsymbol{\tau}_1, \boldsymbol{T}_{\beta,\phi}\boldsymbol{\tau}_2)$. Using (13) and multiplying matrices $\boldsymbol{T}_{\alpha,\theta}$ and $\boldsymbol{T}_{\beta,\phi}$ we get,

$$\begin{aligned} \tilde{y}_3(\rho, \phi; \boldsymbol{\tau}_1, \boldsymbol{\tau}_2) &= x_3(\boldsymbol{T}_{\alpha,\theta}\boldsymbol{T}_{\beta,\phi}\boldsymbol{\tau}_1, \boldsymbol{T}_{\alpha,\theta}\boldsymbol{T}_{\beta,\phi}\boldsymbol{\tau}_2) \\ &= x_3(T_{\alpha\beta,\phi+\theta}\boldsymbol{\tau}_1, T_{\alpha\beta,\phi+\theta}\boldsymbol{\tau}_2) \\ &\triangleq \tilde{x}_3(\rho + \log\alpha, \phi + \theta; \boldsymbol{\tau}_1, \boldsymbol{\tau}_2). \end{aligned}$$

Proposition 2 establishes the equivalence of the rotation and/or scaling of the original 2-D signal with a two-dimensional shift in the $\tilde{x}_3(\rho, \phi; \boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ domain with respect to ρ and ϕ. The following proposition establishes the inverse relation.

Proposition 3: If (17) holds for all classes $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ with the *same values of* α and θ, then, $y(\boldsymbol{t})$ can be generated from $x(\boldsymbol{t})$ by rotation (θ), rescaling (α) and any arbitrary translation.

Proof: Analogous to the proof of Proposition 2.

The following Proposition 4 defines the non-redundant region of support for $\boldsymbol{\tau}_1, \boldsymbol{\tau}_2$, such that $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ are (mutually) distinct.

Proposition 4: All essentially distinct classes $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$, i.e., classes that are not rearrangements of each other can be generated, fixing $\boldsymbol{\tau}_1$ to the unit vector of the horizontal axis of $\boldsymbol{R}^2$, $\tau_0 = [1, 0]$, and varying $\boldsymbol{\tau}_2$ in the zone $S_0 \triangleq [1, 0] \times [0, \infty)$.

Proof: See Appendix A.

Propositions 2.2 and 2.3 suggest that *any shift-invariant transformation of the classes $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ w.r.t. ρ and ϕ that takes care of the alignment between the shifting of different classes* is a representation of 2-D signals that has properties [P1], [P2].

In the sequel we introduce such a transform, which is not the only one that satisfies the aforementioned requirements, but it is very convenient for implementation purposes.

Let

$$\tilde{X}_3(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) \triangleq \int_0^{\infty}\int_{-\pi}^{\pi} \tilde{x}_3(\rho,\phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)\exp\left(-\rho\mathrm{P}-\frac{1}{2\pi}\phi\Phi\right)d\rho d\phi. \tag{18}$$

be the 2-D Fourier transform of the field $\tilde{x}_3(\rho,\phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ with respect to the "space" variables ρ and ϕ. We define next the following collection of 2-D signals/features as a transformed representation of the original signal $x(\boldsymbol{t})$ with the properties [P1], [P2].

Definition:

$$F_x \triangleq \left\{ s_x|\tilde{X}_3(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)|, H(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2), \frac{\partial^2 \mathrm{A}_x}{\partial\tau\partial\mathrm{P}}, \frac{\partial^2 \mathrm{A}_x}{\partial\tau\partial\Phi}\right\}, \tag{19}$$

with,

$$s_x(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) \triangleq \begin{cases} \mathrm{sgn}\left(\tilde{X}_3(\mathbf{o},\mathbf{o};\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)\right) & \mathrm{P}=0,\Phi=0 \\ 1 & \text{elsewhere} \end{cases},$$

where s_x retains information about the sign of the real number $\tilde{X}_3(\mathbf{o},\mathbf{o};\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$,

$$\mathrm{A}_x(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) \triangleq \arg\tilde{X}_3(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2),$$

and,

$$H(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) \triangleq \begin{bmatrix} \frac{\partial^2 \mathrm{A}_x}{\partial \mathrm{P}^2} & \frac{\partial^2 \mathrm{A}_x}{\partial \mathrm{P}\partial\Phi} \\ \frac{\partial^2 \mathrm{A}_x}{\partial\Phi\partial\mathrm{P}} & \frac{\partial^2 \mathrm{A}_x}{\partial\Phi^2} \end{bmatrix},$$

is the Hessian matrix of $\mathrm{A}_x(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ w.r.t. P and Φ and $\tau = [\boldsymbol{\tau}_1^t,\boldsymbol{\tau}_2^t]^t$ parametrizes the set of all possible classes $C(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$. In (19) and elsewhere the indices $(\mathrm{P},\Phi,\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ of A_x have been omitted for notational simplicity. The following theorem states that F_x possesses properties [P1-P2].

Theorem 1: $F_x = F_y$ iff $there is \boldsymbol{T},\boldsymbol{t}_0 : y(\boldsymbol{t}) = x(\boldsymbol{T}\boldsymbol{t}+\boldsymbol{t}_0)$.

Proof:

1) Let $y(\boldsymbol{t}) = x(\boldsymbol{T}\boldsymbol{t}+\boldsymbol{t}_0)$ where

$$\boldsymbol{T} = \alpha\begin{bmatrix}\cos\theta & -\sin\theta \\ \sin\theta & \cos\theta\end{bmatrix};$$

according to Proposition 2 $\tilde{y}_3(\rho,\phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) = \tilde{x}_3(\rho+\log\alpha,\phi+\theta;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$. Hence,

$$\tilde{Y}_3(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) = \tilde{X}_3(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)\exp\left\{j\left(\mathrm{P}\log\alpha+\frac{1}{2\pi}\Phi\theta\right)\right\}.$$

due to the shift property of the Fourier transform w.r.t. variables ρ and ϕ. Consequently, $s_x(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) = s_y(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ and

$$s_x|\tilde{Y}_3(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)| = s_y|\tilde{X}_3(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)|(fa\mathrm{P},\Phi,\boldsymbol{\tau}_1,\boldsymbol{\tau}_2. \tag{20}$$

On the other hand, if $\Delta\mathrm{A}(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) \triangleq \mathrm{A}_y(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) - \mathrm{A}_x(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ we get,

$$\frac{\partial\Delta\mathrm{A}}{\partial\mathrm{P}} = \log\alpha \text{and} \frac{\partial\Delta\mathrm{A}}{\partial\Phi} = \frac{1}{2\pi}\theta. \tag{21}$$

Equation (21) shows that the Jacobian of A_y is not equal to the Jacobian of A_x, but their difference depends on the values of α and θ. For this reason we use the Hessian matrix H, defined in (19), to obtain equality of the two measures. It can be easily verified that,

$$\frac{\partial^2\Delta\mathrm{A}}{\partial\mathrm{P}^2} = \frac{\partial^2\Delta\mathrm{A}}{\partial\Phi\partial\mathrm{P}} = \frac{\partial^2\Delta\mathrm{A}}{\partial\tau\partial\mathrm{P}} = 0\text{and} \tag{22}$$

$$\frac{\partial^2\Delta\mathrm{A}}{\partial\Phi^2} = \frac{\partial^2\Delta\mathrm{A}}{\partial\mathrm{P}\partial\Phi} = \frac{\partial^2\Delta\mathrm{A}}{\partial\tau\partial\Phi} = 0, \tag{23}$$

which completes the proof of the necessity part.

2) Let, now, $F_x = F_y$. Clearly (22) and (23) hold. Since the Hessian,

$$\Delta H = \begin{bmatrix} \frac{\partial^2\Delta\mathrm{A}}{\partial\mathrm{P}^2} & \frac{\partial^2\Delta\mathrm{A}}{\partial\Phi\partial\mathrm{P}} \\ \frac{\partial^2\Delta\mathrm{A}}{\partial\mathrm{P}\partial\Phi} & \frac{\partial^2\Delta\mathrm{A}}{\partial\Phi^2} \end{bmatrix} = 0, \tag{24}$$

we infer that there exist $\rho_{\boldsymbol{\tau}_1,\boldsymbol{\tau}_2}$, $\phi_{\boldsymbol{\tau}_1,\boldsymbol{\tau}_2}$, $\lambda_{\boldsymbol{\tau}_1,\boldsymbol{\tau}_2}$ which are constant for a specific class $\boldsymbol{\tau}_1$, $\boldsymbol{\tau}_2$ such that

$$\Delta\mathrm{A}(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) = \rho_{\boldsymbol{\tau}_1,\boldsymbol{\tau}_2}\mathrm{P} + \phi_{\boldsymbol{\tau}_1,\boldsymbol{\tau}_2}\Phi + \lambda_{\boldsymbol{\tau}_1,\boldsymbol{\tau}_2}. \tag{25}$$

Employing the last equalities of (22) and (23) we conclude that $\rho_{\boldsymbol{\tau}_1,\boldsymbol{\tau}_2}$ and $\phi_{\boldsymbol{\tau}_1,\boldsymbol{\tau}_2}$ are independent of $\boldsymbol{\tau}_1$, $\boldsymbol{\tau}_2$, i.e., there exist ρ_0 and ϕ_0 such that:

$$\Delta\mathrm{A}(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) = \rho_0\mathrm{P} + \phi_0\Phi + \lambda_{\boldsymbol{\tau}_1,\boldsymbol{\tau}_2}. \tag{26}$$

Equation (26) for $\mathrm{P}=0$, $\Phi=0$ takes the form,

$$\Delta\mathrm{A}(\mathbf{o},\mathbf{o};\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) = \lambda_{\boldsymbol{\tau}_1,\boldsymbol{\tau}_2}. \tag{27}$$

Since however, $\tilde{X}_3(\mathbf{o},\mathbf{o};\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ and $\tilde{Y}_3(\mathbf{o},\mathbf{o};\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ are real numbers,

$$\lambda_{\boldsymbol{\tau}_1,\boldsymbol{\tau}_2} = 0(mod\pi). \tag{28}$$

Taking into account that also

$$s_y|\tilde{Y}_3(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)| = s_x|\tilde{X}_3(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)| \tag{29}$$

we infer first that $\lambda_{\boldsymbol{\tau}_1,\boldsymbol{\tau}_2} \equiv 0$. (since s_x will be equal to s_y for all $(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$) and we conclude next that

$$|\tilde{Y}_3(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)| = |\tilde{X}_3(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)|. \tag{30}$$

As a consequence, $\tilde{y}_3(\rho,\phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ is simply a shift of $\tilde{x}_3(\rho,\phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ by (ρ_0,ϕ_0), i.e.,

$$\tilde{y}_3(\rho,\phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) = \tilde{x}_3(\rho+\rho_0,\phi+\phi_0;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2). \tag{31}$$

Equation (31) in view of Proposition 3 completes the proof of the sufficiency part.

The above theorem proves that indeed the proposed representation F_x fulfills the desired requirements ([P1-P2]).

If multiple independent records of $x(\boldsymbol{t})$ are available, noise insensitivity can be achieved by performing ensemble averaging of the third-order correlation domain (c.f. (8) and related discussion). F_x will then be resistant to any type of non-skewed additive noise (e.g., Gaussian, even spatially correlated). Under this condition, property [P3] is also satisfied. Multiple independent records are also used for the same purpose in [2], [3], [31] and [32]. In Section III it is further

shown that improvement of the signal-to-noise ratio can be obtained even in the single record case using appropriate local averaging of the triple-correlation representation.

It should be emphasized that F_x is a *stand-alone* representation which can be used as a direct input to a neural network classifier, as will be described in Section IV; this is in contrast to other SRS representations that require a *matching* procedure of the pattern to be classified with all available prototypes.

III. Reducing the Size of the Representation-Discretizations

A. From Uniqueness to Sufficiency

When comparing two 2-D signals, $x(\boldsymbol{t})$ and $y(\boldsymbol{t})$, using either a neural network or a conventional classifier, the matching of the first two components of F_x and F_y, namely $s_x|\tilde{X}_3(P,\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)|$ with $s_y|\tilde{Y}_3(P,\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)|$ and $H_x(P,\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ with $H_y(P,\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$, checks whether in the interior of each class $C(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ the field $\tilde{x}_3(\rho,\phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ is a shift of $\tilde{y}_3(\rho,\phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$. If this is true, the matching of the last two components of F_x, F_y guarantees that the shifting is the same for all classes. If both requirements are met, Theorem 1 states that $y(\boldsymbol{t})$ coincides with $x(\boldsymbol{t})$ within a shift-rotation and rescaling.

Although one can implement the matching procedure in a straightforward way, using standard techniques for discretizing the components of $F_x(F_y)$ and approximating the derivatives in a digital manner by differences, the resulting computational load would be extremely high. For this reason, the following three representations of reduced size are considered in the rest of the paper.

1) As a first reduction of the problem's size, we propose to drop out the last two components of F_x, F_y considering that it is highly unlikely to face a situation where *all* classes $C(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ of $y(\boldsymbol{t})$ are shifts of the corresponding classes of $x(\boldsymbol{t})$ and still $F_x \neq F_y$. Especially in applications where the recognition algorithm attempts to classify a given pattern in a set of finitely many prototypes, the above situation becomes even more improbable.
2) As a second reduction step, we propose to keep only the first component of F_x (F_y), i.e., keep only the amplitude information of the Fourier transform in (18). An additional evidence, for dropping out matrix H from our representation, is that, under certain conditions, the amplitude $|\tilde{X}_3(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)|$ of the Fourier transform $\tilde{X}_3(\mathrm{P},\Phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ includes all the necessary information for re-synthesizing $\tilde{x}_3(\rho,\phi;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$, (see [15], [24] and [34]).
3) Finally, further reduction can be achieved if the 0th frequency coefficient $\tilde{X}_3(0,0;\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ of each class is used as a "sufficient" feature. A similar size reduction can also be found in [31], [32].

In any case, when the pattern space under consideration is predetermined, one can always check up to which point the size reduction of F_x does not ruin the uniqueness of the representation.

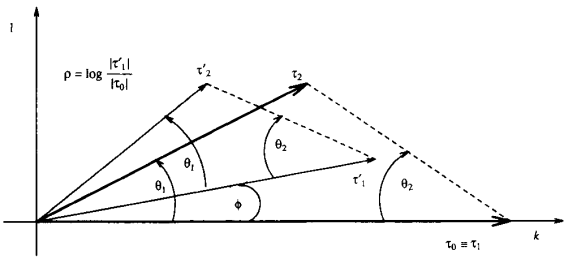


Fig. 3. Input space parametrization using θ_1, θ_2 angles.

B. Parametrization

Due to Proposition 4, the indices $\boldsymbol{\tau}_1,\boldsymbol{\tau}_2$ in the definition of F_x can take the form,

$$(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) = (\tau_0,[k,l])0 \leq k \leq 1, 0 \leq l, \tag{32}$$

An alternative scheme is to parametrize $(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ via the angles θ_1,θ_2 included between the plane vectors $(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ and $(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2-\boldsymbol{\tau}_1)$ respectively, fixing again $\boldsymbol{\tau}_1$ to τ_0 (see Fig. 3), i.e.,

$$(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2) = (\tau_0,\boldsymbol{\tau}_2) \rightarrow (\theta_1,\theta_2)\theta_1,\theta_2 \in \left[0,\frac{\pi}{2}\right]. \tag{33}$$

It is interesting to mention that either of the above schemes lead to considerable reduction of the invariant representation input space. In particular, the four dimensional space spanned by $(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ is reduced, without any loss of information, to a 2-D space that is the $S_0 = [1,0] \times [0,\infty)$ region, in the first scheme, and the $[0,\pi/2] \times [0,\pi/2]$ in the second scheme.

C. Discrete Implementation

In real-life applications, the original 2-D signal $x(\boldsymbol{t})$ is available in discrete form; i.e., $\boldsymbol{t} \in [0,\cdots,N-1] \times [0,\cdots,N-1]$. In that case the triple-correlation can be computed via (4), giving rise to a 4-D representation $x_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$. Computationally, it is more preferable to obtain $x_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$ as the inverse FFT of the bispectrum $X_3(\boldsymbol{u},\boldsymbol{v})$, (the Fourier transform of $x_3(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$) (see e.g., [23] for details). This is so, because

$$X_3(\boldsymbol{u},\boldsymbol{v}) = X(\boldsymbol{u})X(\boldsymbol{v})X(-\boldsymbol{u}-\boldsymbol{v}). \tag{34}$$

where $X(\boldsymbol{u})$ is the 2-D Fourier transform of $x(\boldsymbol{t})$. As a consequence, $X_3(\boldsymbol{u},\boldsymbol{v})$ can be computed as the triple product of a 2-D FFT using fast software or hardware implementations.

The next step is to use one of the parametrization schemes of the previous subsection, in order to specify a discrete 2-D grid of (k,l) in (32) or (θ_1,θ_2) in (33), that will determine the number of distinct classes $C(\boldsymbol{\tau}_1,\boldsymbol{\tau}_2)$. The number of possible values that (k,l), or (θ_1,θ_2), take, is specified solely by the dimension N of the input signal. In general, however, these values should be quantized to a rather small number of levels, defining the effectively distinct classes. Clearly, the quantization of (k,l), or (θ_1,θ_2), on a discrete grid results in a possible loss of information; the coarser the quantization is chosen, the greater the resulting loss of information is.

After deciding which distinct classes are to be used, another discretization should be applied in the interior of each class.

The field $\tilde{x}_3(\rho,\phi;\tau_1,\tau_2)$ should be computed on a discrete grid of the parameters ρ and ϕ. The sampling rate in this domain is conceptually related to the number of triple-correlation lags that are assigned to each class; in general for an $N \times N$ input signal, the total number of possible correlation lags is $O(N^4)$. Consequently, there is a trade-off between the number of effectively distinct classes and the sampling rate of this phase. As a rule of thumb, the resulting grid should be able to accommodate, in separate positions, a significant portion of lags $x_3(\tau_1,\tau_2)$ assigned to $C(\tau_1,\tau_2)$ and simultaneously be parsimonious enough, to avoid generation of sparse fields $\tilde{x}_3(\rho,\phi;\tau_1,\tau_2)$

Having available $\tilde{x}_3(\rho,\phi;\tau_1,\tau_2)$ for each class $C(\tau_1,\tau_2)$, in the form of a 2-D matrix, an FFT algorithm can be used to compute $\tilde{X}_3(\mathrm{P},\Phi;\tau_1,\tau_2)$. Since there is no interrelation between different classes, a parallel implementation of these computations is possible.

In this discrete framework, the representation F_x of $x(t)$ can be easily computed from $\tilde{X}_3(\mathrm{P},\Phi;\tau_1,\tau_2)$, by considering the amplitude of this signal $|\tilde{X}_3(\mathrm{P},\Phi;\tau_1,\tau_2)|$, for all P, Φ not equal to zero and $\tilde{X}_3(0,0;\tau_1,\tau_2)$, for $\mathrm{P}=\Phi=0$, and by approximating the included derivatives with differences of adjacent terms of $\arg\tilde{X}_3(\mathrm{P},\Phi;\tau_1,\tau_2)$.

D. Effects of Quantization

Although quantization of (θ_1,θ_2) for the determination of distinct classes as well as quantization of (ρ,ϕ) in the interior of each class, has the drawback of losing some information, it has two interesting side-effects. Namely, enhances the robustness of the proposed representation against local distortions and also provides noise insensitivity, even if the representation is computed based on a single data record.

a) Small local distortions of an object cause shift of the triple-correlation lags that are included in each class defined by, say, (θ_1,θ_2) to the neighboring classes defined by $(\theta_1 \pm \epsilon, \theta_2 \pm \epsilon)$, for a number of small values of ϵ. As a consequence, quantization that is equivalent to local averaging of the image representation, using windows of size $(\epsilon \times \epsilon)$ on the (θ_1,θ_2) space, can alleviate this undesired effect.

b) In the quantized version of the representation,

$$\tilde{x}_3(\rho,\phi;\theta_1,\theta_2) = \frac{1}{L}\sum_{A} x_3(\tau_1,\tau_2). \tag{35}$$

where A contains all indices (τ_1,τ_2) which form a triangle with angles $(\theta_1 \pm \epsilon, \theta_2 \pm \epsilon)$, with $|\tau_1| = |\tau_0|(\rho \pm \Delta\rho/2)$ and $\tau_1\hat{\tau}_2 = \phi \pm \Delta\phi/2$ where $\Delta\rho$ and $\Delta\phi$ are the quantization steps in the ρ,ϕ domain. The cardinal number of A increases when quantization is chosen to get coarser and coarser and when the object size $T \to \infty$. In view of (10), (35) results in the suppression of non-skewed additive noise, at the cost of degraded resolution.

F. Special Cases

In the spirit of the first part of this section, the three reduced formulations of F_x can be easily implemented as follows.

The first representation has the form,

$$\begin{aligned} F_x &= \left\{ s_x|\tilde{X}_3(\mathrm{P},\Phi;\tau_1,\tau_2)|, H_x(\mathrm{P},\Phi;\tau_1,\tau_2)\right\}_{\substack{\mathrm{P}=0\cdots R-1\\ \Phi=0\cdots K-1}} \\ &= \left\{ s_x|\tilde{X}_3(\mathrm{P},\Phi;\theta_1,\theta_2)|, H_x(\mathrm{P},\Phi;\theta_1,\theta_2)\right\}_{\substack{\theta_1=0\cdots L_1-1,\\ \theta_2=0\cdots L_2-1}} \end{aligned} \tag{36}$$

where $s_x|\tilde{X}_3(\mathrm{P},\Phi;\tau_1,\tau_2)|$ and each of the four elements of the matrix $H_x(\mathrm{P},\Phi;\tau_1,\tau_2)$ consists of a 2-D $(R \times K)$ signal for each of the $L_1 \times L_2$ classes; or equivalently of a 2-D signal of size $(RL_1 \times KL_2)$, where R,K, L_1 and L_2 denote the number of levels used for quantizing variables P,Φ,θ_1 and θ_2 respectively, the latter variables being assumed to be normalized w.r.t. the corresponding quantization steps.

The second representation is defined as,

$$\begin{aligned} F_x &= \left\{ s_x|\tilde{X}_3(\mathrm{P},\Phi;\tau_1,\tau_2)|\right\}_{\substack{\mathrm{P}=0\cdots R-1\\ \Phi=0\cdots K-1}} \\ &= \left\{ s_x|\tilde{X}_3(\mathrm{P},\Phi;\theta_1,\theta_2)|\right\}_{\substack{\theta_1=0\cdots L_1-1,\\ \theta_2=0\cdots L_2-1}} \end{aligned} \tag{37}$$

where only one 2-D signal of size $(RL_1 \times KL_2)$ is considered.

The third representation is

$$\begin{aligned} F_x &= \left\{ s_x|\tilde{X}_3(0,0;\tau_1,\tau_2)|\right\} \\ &= \left\{ \tilde{X}_3(0,0;\theta_1,\theta_2)\right\}_{\theta_2=0\cdots L-1}^{\theta_1=0\cdots L-1}, \end{aligned} \tag{38}$$

where only a 2-D $(L_1 \times L_2)$ signal is considered.

In summary, given a zero-mean input 2-D image, its triple-correlation is first efficiently calculated through the bispectrum computation; straightforward clustering of the triple-correlation lags in invariant classes is then performed, using a look-up table for a given image size. Using a 2-D FFT, in parallel, for each class, all elements in F_x are computed, so that the resulting input representation can be used for SRS invariant classification.

IV. Neural Network Architectures for Invariant Classification

Based on the invariant third-order correlation representations, derived in the previous section, neural network architectures are proposed in the following, as an efficient, invariant classification scheme.

A. Efficient Implementation of Third-Order Networks

Let us consider a single-layer third-order neural network, shown in Fig. 1, the input of which is a two dimensional image. Let us also assume that backpropagation is used to train the network, by minimizing the sum of squared error between the desired and actual outputs of neurons in the output layer. Following the discussion given in the Introduction of the

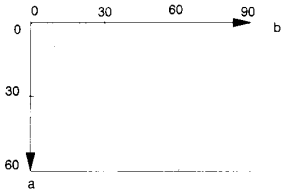


Fig. 4. Input space representation in θ_1. θ_2 plane with $\theta_1 \leq \theta_2$.

paper, the updating of the weights w_{ijkl}, connecting triplet input products to output neurons, should be performed under the constraint given in (2). Introducing this constraint in the computation of y_i in (1), the following expression is obtained

$$y_i = f\left[\sum_{a,b,c} w_{iabc} x_{abc}\right]. \tag{39}$$

$$x_{abc} = \sum_{j'k'l'} x_{j'k'l'}. \tag{40}$$

where variables j', k', l' refer to all input triplets, forming triangles with angles a,b,c. Equation (39) can be viewed as describing the operation of a first-order neuron, the inputs of which correspond to classes (abc) of triangles that are similar to each other.

Proposition 5: Third-order neural networks are a specific category of triple-correlation-based neural networks, in the sense that the input to the former networks is equivalent to the special-case representation given in (38) of Section III.

Proof: It is a consequence of (40), (38) and definitions (16) and (4); the property that the zeroth sample of the Fourier transform of a signal is equal to the sum of the signal samples should be used for this purpose, with angles θ_1, θ_2 in (33) corresponding to angles a,b of the triangles in (40). Due to (33), angle a should be chosen as an acute angle of the triangle in (40), which has in its right-hand-side, in the anti-clockwise direction, another acute angle, i.e., angle b.

Equation (40) can be viewed as transforming the original input image space to a 2-D (a, b) image space. For simplicity, let us regard angles a, b, c in the following order of magnitude

$$a \leq b \leq c. \tag{41}$$

and select angles a and b to define the 2-D invariant image representation. Adopting notation (41), the order of scanning the triangles is ignored, resulting in ignoring reflection of the image plane as one of possible input transformations. In this case, the (a, b) input space further reduces to $(0, \pi/3) \times (0, \pi/2)$ as shown in Fig. 4.

The number of inputs x_{ab} (or equivalently x_{abc}) in (39) is equal to the number of possible triangle classes; it, therefore, depends on the input image size. Table I verifies that the number of possible classes in (a, b) increases with the size of the input image; an excessive number of inputs can be therefore generated in a real-life application. As a consequence, the third-order neural network or equivalently the first-order neural network described in (39) may be of impractical size. For this reason, quantization or local averaging of the (a, b) input grid, as was described in the previous section, can effectively reduce the problem size, providing also insensitivity to distortions of the input image and enhancing insensitivity to additive noise.

TABLE I
NUMBER OF CLASSES FOR $(N \times N)$ IMAGES $N = 2, \cdots, 23$.

Image size	Number of classes
2 × 2	1
3 × 3	7
4 × 4	21
5 × 5	56
6 × 6	120
7 × 7	230
8 × 8	403
9 × 9	668
10 × 10	1020
11 × 11	1537
12 × 12	2217
...	
16 × 16	7204
17 × 17	9297
18 × 18	11 756
...	
23 × 23	31 959

B. Triple-Correlation-Based Neural Network Classifiers

All invariant representations given by (36)–(38) in Section III, consist of 2-D images of various sizes. The last one, which has been used above to derive an efficient implementation of third-order neural networks, is a 2-D $(L_1 \times L_2)$ image, where L_1 and L_2 are the numbers of quantization levels for angles θ_1 and θ_2 respectively. For small sizes of L_1 and L_2, it is possible to apply a multilayer fully-connected neural network classifier as shown in Fig. 5, which is trained by some efficient back-propagation variant. For larger sizes however, the number of free-parameters (i.e., the number of interconnection weights) increases rapidly, imposing problems on the generalization ability of the network. Weight decay [4], [14] during training, is a technique, which can adapt the network size, so as to avoid problems like over-fitting. Various studies can be found in the literature comparing the performance of neural network classifiers to conventional statistical ones. Recent studies concerning the problem of image classification/recognition, either in the case of handwritten image data [20], [6], or in generic and real image data cases [35], indicate that structured neural network classifiers are especially useful when applied directly to image pixel values, and not to a set of features extracted from images. It is in this case that neural networks, with their massive parallelism and distributed processing, can outperform any type of conventional statistical classifiers, such as the Bayes, K-nearest neighbor and minimum-measure distance ones. The need for structured networks, trained by appropriate learning algorithms, is more evident in the next case, where representation (37) of the previous section is used for classification.

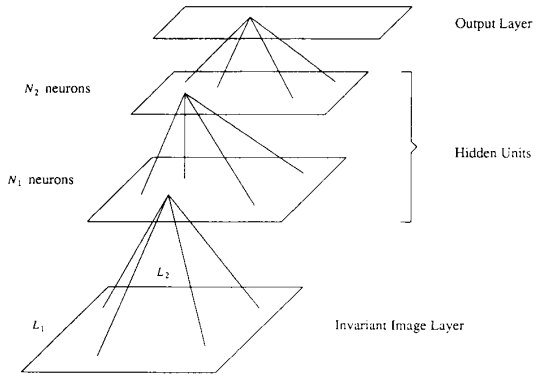


Fig. 5. A neural network classifier based on invariant representation (38).

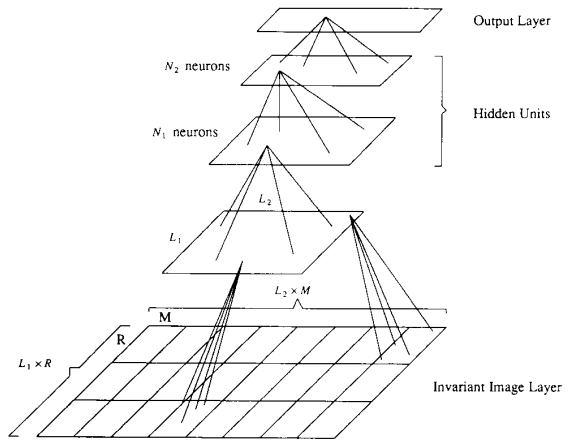


Fig. 6 A neural network classifier based on invariant representation (37).

In the representation described by (37), a 2-D image of size $(R, M)_{\theta_1,\theta_2}$ corresponds to each class defined by (θ_1, θ_2). Let us, for simplicity, consider that all these 2-D images are of the same size, say $(R \times M)$. Then, the resulting invariant representation is a 2-D image of size $(L_1 \times R, L_2 \times M)$, in which neighboring classes correspond to neighboring sub-images of size $(R \times M)$. A neural network architecture, that can be used for classification of this input representation, is a receptive-field multilayered network. A minimal configuration of such a network is shown in Fig. 6. The first hidden layer of this network is in the form of a 2-D $(L \times L_2)$ image, each pixel (neuron) of which has a corresponding receptive field on a sub-image of size $(R \times M)$ of the invariant image representation. In general, the neurons of this hidden layer correspond to the different classes of the input representation. The following hidden layers of the network can have, either a receptive field or a fully-connected structure, depending on the size of the first hidden layer image.

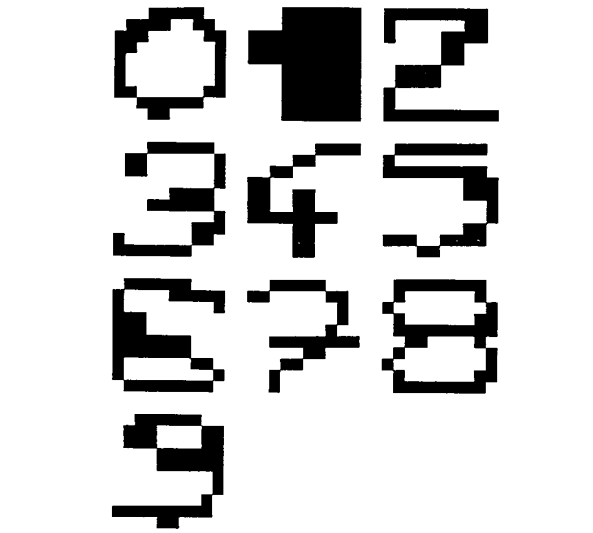
In the invariant representation described by (36), which includes not only amplitude, but also phase information, the input to the network consists of five such $(L_1 \times R, L_2 \times M)$ images; one corresponding to the previous case, where amplitude information was considered, and four containing the Hessian of the phase information. Efficient variants of backpropagation [16], [18], [21], combined with weight decay, can be used to train the different receptive field network architectures, resulting in a scheme for invariant classification of the input images that is also insensitive to additive noise and to the existence of small distortions.

V. Simulation Results

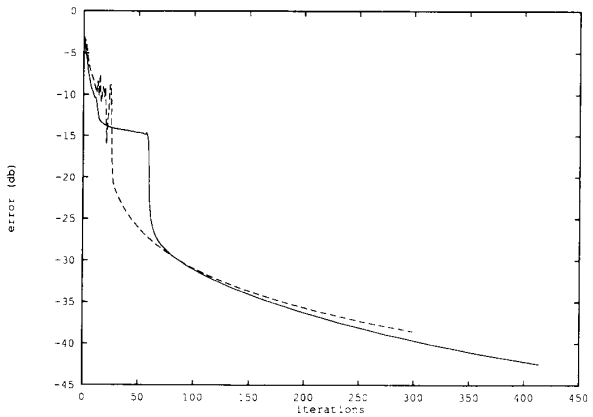
The performance of the neural network classifiers described in the previous section has been examined for invariant classification of synthetic and real image data. The third-order correlation representation, derived in Sections II and III were used for this purpose.

Test Case 1: (i) The invariant representation described by (38) of Section III, which can also be used for an efficient implementation of third-order neural network classifiers, was investigated first, using a characteristic example; this is handwritten data recognition, which has been a subject of extensive research in the field of image recognition using neural networks (see for example [6], [9], [20]). For the purposes of this paper, a set of data was created, composed of examples of numerals represented as binary images of size (10×10) pixels. Figure 7(a) shows a characteristic sequence of such examples. The representation of these numerals in the (θ_1, θ_2) space, with $0 \leq \theta_1 \leq 60$ degrees and $0 \leq \theta_2 \leq 90$ degrees, using a quantization step of one degree was obtained first, resulting in 2-D gray-scale invariant numeral representations of size (60×90) pixels. A coarser quantization was obtained using local averaging, in windows of (6×6) pixels, reducing the size of the derived images to (10×15) pixels; due to the triangle form of the input space, shown in Fig. 4, the effective number of pixels has been reduced from 2791 in the (60×90) image to 91 in the (10×15) image. This number can be compared to the much greater number (i.e., 1020) of possible distinct classes - inputs to a third-order neural network, given in Table I for a size (10×10) of the input image. Moreover, this input size is comparable to the size (10×10) of the original binary images shown in Fig. 7(a). The invariant representations corresponding to the numerals of Fig. 7(a) are shown in Fig. 8. It is easily seen that these representations are visually different.

(ii) We then examined the ability of a single-hidden-layer fully-connected network with 14 hidden and 10 output units, trained by backpropagation, to learn a training set of two hundred such images. For comparison purposes, identical networks were used to learn the set of binary images, as the ones shown in Fig. 7(a), and the set of corresponding invariant representations. Figure 7(b) compares the learning speed of the networks in the two cases, measured in terms of the logarithm of the minimized error function, with respect to the number of iterations (passes through the data set). It can be easily seen that convergence was similar in both cases. This indicates that the proposed representation, although invariant with respect to input image transformations, does not require a more complex network classifier than the one needed for the case of original images.



(a)

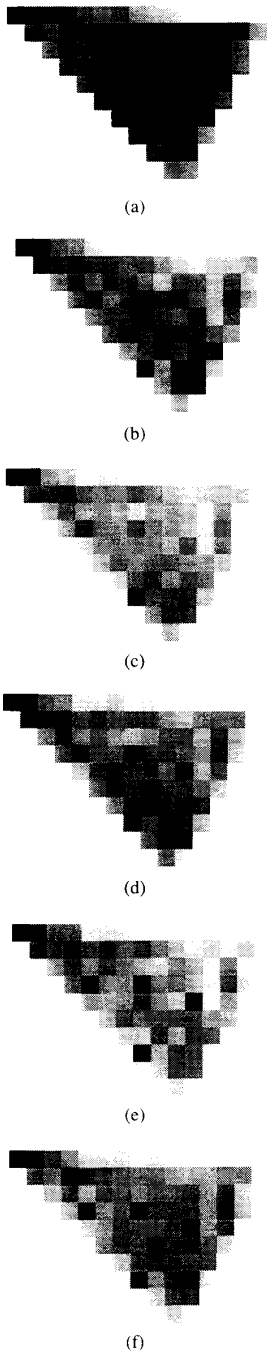


(b)

Fig. 7. (a) A set of hand-written numerals (0–9). (b) Convergence of the learning algorithm when using original image data (continuous line) or invariant representations (dashed line).

In order to further illustrate the robustness of the derived representations with respect to distortions of the input image, we included in Fig 9(a)–(h) four (16×16) gray scale images showing digits "4" and "9" and their corresponding invariant representations. It can be easily seen, that the representations of digit "9" are very similar, despite the small differences in the original images; the representations of digit "4" are also similar, despite the differences in the original images, being also easily discerned from the representations of digit "9."

Test Case 2: Having used this synthetic example to illustrate the invariant image representations obtained in the case of a well known problem, we have chosen a real-life application to examine the performance and the generalization of the proposed triple-correlation-based neural network classifiers. Inspection of solder joints in printed circuit board manufacturing was chosen as such an application in which conventional pattern recognition techniques have not shown



(a) (b) (c) (d) (e) (f)

Fig. 8. Invariant representations (a)–(j) corresponding to numerals (0–9).

sufficient reliability. 2-D gray-scale images, showing either the height or the intensity as functions of the position across solder joints, were obtained by an optical laser scanner and used as signals to be classified in two categories; namely, good or poor solder joints, the latter containing insufficient amount of solder. An example of each category is shown in Fig. 10. Invariance of classification, with respect to input image transformations, as well as insensitivity to additive

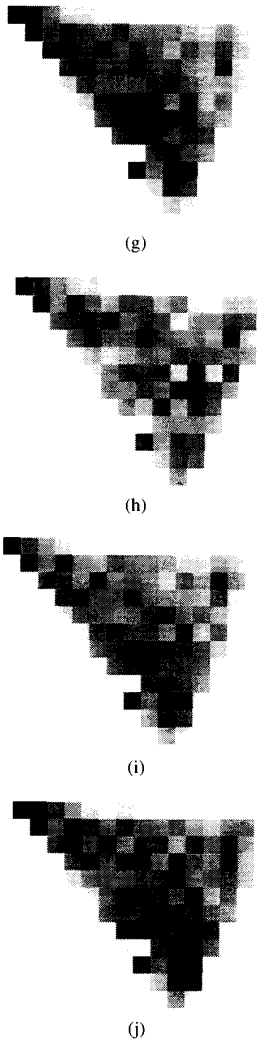


Fig. 8. *Continued.*

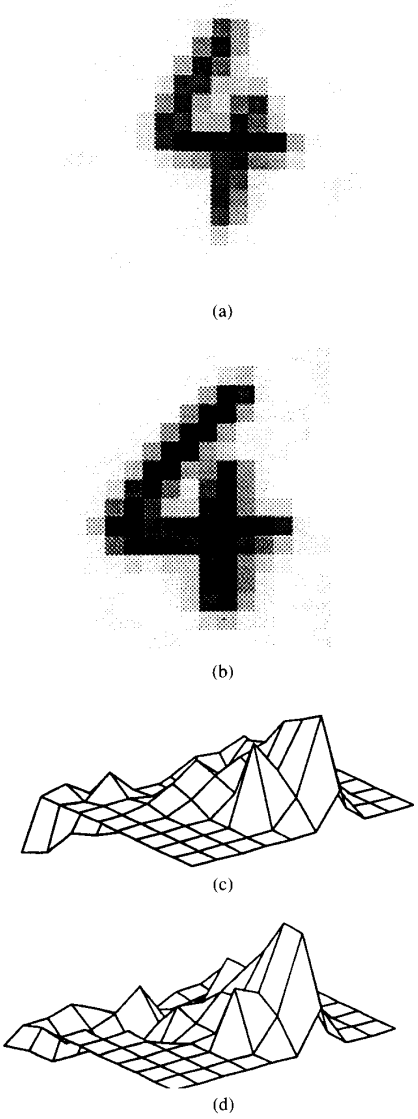


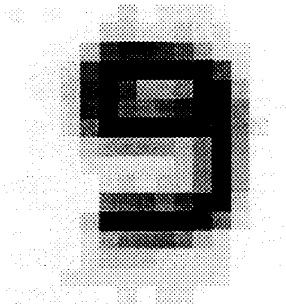
Fig. 9. Gray-scale images of digits "4" and "9" and the corresponding invariant representations (a)–(h).

noise and distortions are important aspects of this problem. We next examined the performance of both types of neural network classifiers shown in Figs. 5 and 6 of the previous section, which are based on the corresponding invariant image representations.

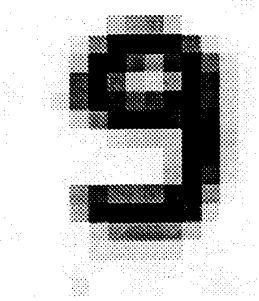
Test Case 2a: (i) Figure 11 shows the invariant representations of the images of Fig. 10, derived using formula (38) in the (τ_1, τ_2) domain. The values of (τ_1, τ_2) were quantized, as in the previous simulations, to a discrete grid of (60×90) pixels. These representations are quite different from each other; they can, therefore, be used as inputs to the network classifier. To illustrate the noise insensitivity of the derived representations, a noisy version of the "good" solder joint image shown in Fig. 10, was generated next, by adding zero-mean white Gaussian noise to the signal, at a signal-to-noise-ratio of 10 dB; the resulting image, as well as the corresponding invariant representation are shown in Fig. 12(a) and 12(b). It can be easily seen that the derived representation does not differ from the one shown in Fig. 11(a). Moreover, to illustrate invariance with respect to input transformations, such as rotation, a solder joint image and a version of it rotated by 45 degrees are shown, together with the corresponding representations in the (τ_1, τ_2) domain, in Fig. 13(a)–(d).

(ii) Local averaging, using windows of various sizes, was used to reduce the size of the grid to (10×10) or (10×15) pixels. This size can also be compared to the number of possible classes, given in Table I, for an image of (23×23) pixels; an excessive number of around 32 000 inputs would be required in a standard third-order neural network for invariant classification of the images.

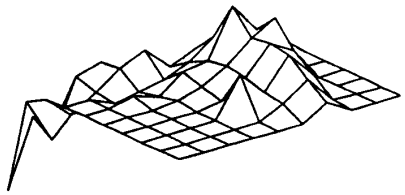
100 input images, 50 of each category, were transformed in this domain and used for training the neural network shown in Fig. 5. A two hidden layer network was used, varying the



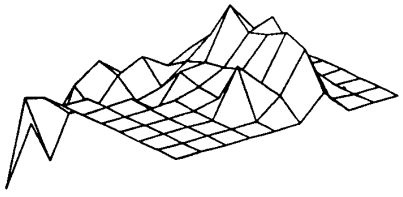
(e)



(f)

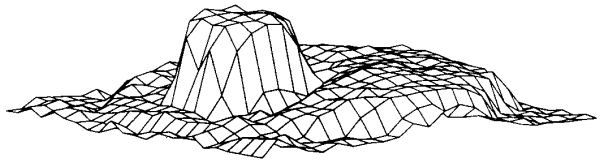


(g)

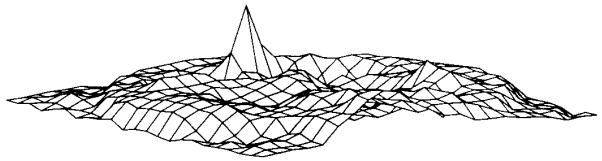


(h)

Fig. 9. *Continued.*

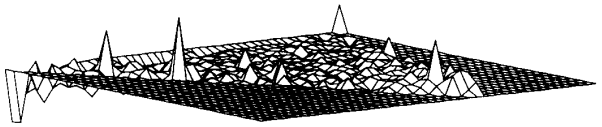


(a)

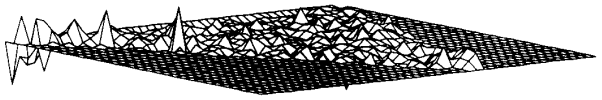


(b)

Fig. 10. (a) An image of a "good" solder joint. (b) An image of a "poor" solder joint.

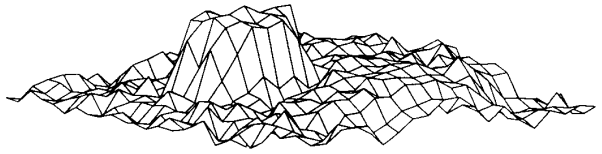


(a)

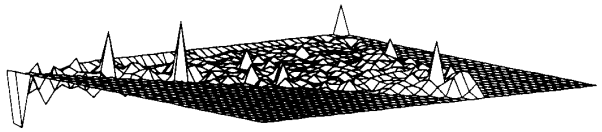


(b)

Fig. 11. Invariant representations corresponding to Fig. 10.



(a)



(b)

Fig. 12. (a) A noisy version of the image in Fig. 10(a). (b) Its invariant representation.

number of neurons in the first hidden layer (i.e., N_1 in Fig. 5) between 3 and 5 and setting the number N_2 of neurons in the second hidden layer to 1 or 2. In all cases the network was able to learn to classify correctly all training data, using backpropagation. A set of 100 different input images, 50 of each category, was used to test the generalization ability of the network; the percentage of correct classifications in the test set was very high, around 94%. A characteristic example is presented next, where the use of a network with $N_1 = 5$ and $N_2 = 1$ hidden neurons is examined and compared in two cases. In the first, the invariant representation of (10×10) pixels was used as the network input; the original image was used in the second case, its size being also reduced to (10×10) pixels through local averaging. Figure 14 shows the percentage of correct classifications in the test data set, using the network interconnection weights computed in each training cycle of the networks. It is easily seen that generalization of the triple-correlation-based network was much better than that of the network based on the original input images (96% compared to 84%).

Test Case 2b: (i) Formula (37) was then used to provide invariant representations of the images shown in Fig. 10, quantizing the range $(-\pi, \pi)$ of each of the (θ_1, θ_2) possible

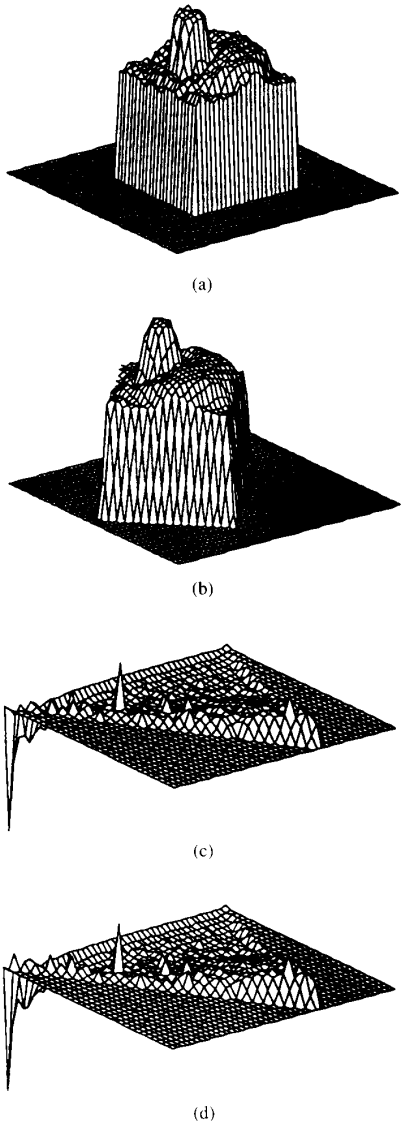


Fig. 13. (a) A good solder joint. (b) A version rotated by 45°. (c) Invariant representation of (a). (d) Invariant representation of (b).

values to $L_1 = L_2 = 10$ levels (giving rise to 100 distinct classes); each of variables P and Φ was uniformly quantized within each class to eight levels. The resulting representations, in the form of 2-D images of size $\{(8 \times 10) \times (8 \times 10)\}$, are shown in Fig. 15(a)–(b). Visual differences are easily discerned between these two representations. Insensitivity to noise and rotation invariance is illustrated, by comparing Fig. 15(a) with Fig. 15(c) and 15(d), which show the representations corresponding to a noisy (SNR = 10dB) and a rotated (by 45 degrees) version of the original image respectively. Figure 16(a)–(d) verifies the above results, by showing the corresponding representations of class (3,1) of the above images in Fig. 15, in the form of 2-D images of size (8×8).

(ii) We then used the same image data sets as above, to train and test the performance of the network classifier

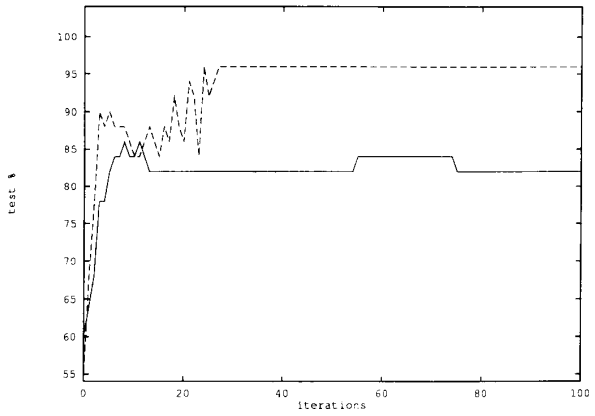


Fig. 14. Generalization ability of the neural network classifier based on (a) original image data (continuous line). (b) Invariant representation (dashed line).

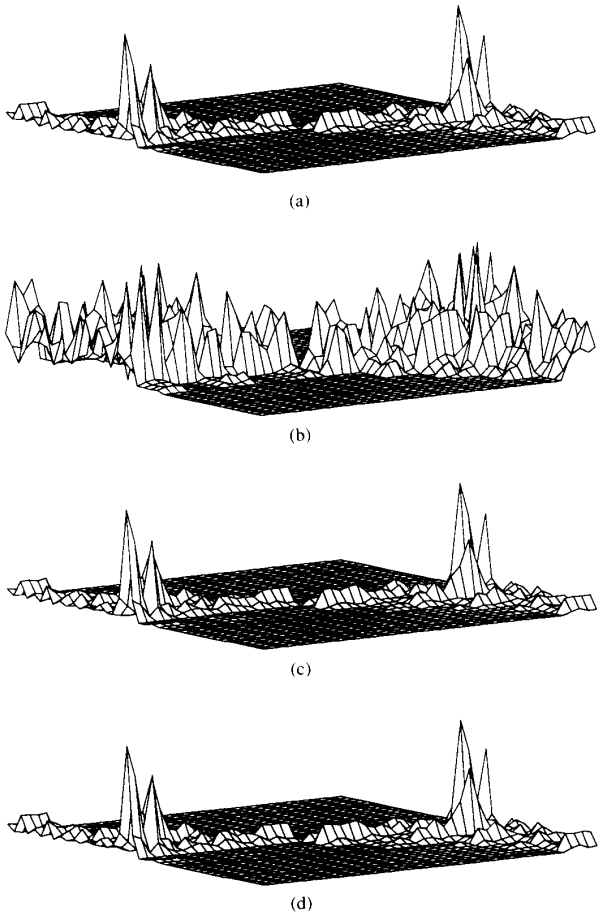


Fig. 15. Invariant representations of the images shown in Fig. 10(a), (a); 10(b), (b); 12(a), (c), a version of 10(a) rotated by 45°; derived using (37).

shown in Fig. 6, with $R = M = 8$, $L_1 = L_2 = 10$, and selecting $N_1 = 5$ and $N_2 = 1$, based on the results of the previous experiment. A second-order backpropagation variant

[18], [21], combined with weight decay was used to train the receptive-field architecture shown in Fig. 6. After training, the generalization ability of the network was raised to 98%, showing that triple-correlation-based networks can be a very effective invariant classification scheme.

Test Case 3: In the last part of our simulations we examined the performance of the proposed scheme to other types of conventional or neural network classifiers.

(i) In the first experiment of this category we compared the performance of conventional classifiers to that of the invariant classifiers used in Test Case 2a(ii), which were derived according to formula (38). The Mahalanobis minimum distance criterion [7] was used to classify the test set of images, by comparing them to two patterns, computed as the mean values of all "good" and "poor" representations of the training set of 100 input images respectively. The classification results were rather poor; only 64% correct classifications were achieved. Moreover, to illustrate the difficulty of the problem and the efficiency of the neural-network-based classification scheme, we also performed an "exhaustive" classification of the test image representations; to do this, we computed the minimum distance of these representations from each one of the 100 training image representations, the latter being considered as known patterns. Even in this case, correct classifications were 92%; i.e., less than the ones provided by the much more computationally efficient neural network classifiers. It should be mentioned that all errors were due to the "poor" solder joint category, which generally includes images that are quite different from each other.

(ii) In the second experiment of this category, we used, as invariant feature, the amplitude of the Fourier transform of the solder joint images after re-sampling them in log-polar form. For each image the cartesian grid was first aligned w.r.t. the corresponding centroid in order to take care of possible shifts. Figures 17(a) and 17(c) depict the representations corresponding to the images of Fig. 10, while the representation of image 10(a) rotated by 45 degrees is given for illustration purposes in Fig. 17(b).

The obtained representations were then presented to the neural network classifiers, used in Test Case 2a. Classification of the same test data set, after training the network with the same training set of images, was only around 82% successful. This verifies the fact that there is much loss of information in the derived representations. In addition, it should be mentioned that this approach gives much deterior performance if the images are contaminated by additive noise; first because the estimation of the centroid is noise sensitive and second, because the Fourier amplitude is affected by even non-skewed type of noise (see also [32]). The registration procedure could be obviated by first computing the amplitude of the 2-D Fourier transform of the images, in order to gain shift insensitivity, and then applying the log-polar transformation to them. In this case there is a further loss of (phase) information, when keeping only the amplitude of the Fourier transform and the resulting representation would still be sensitive to the above types of noise.

(iii) A final test was performed to compare the performance of the proposed triple-correlation-based neural network classifiers to another type of neural network classifiers, based on moment-invariants [17]. The latter technique was applied to the original solder-joint images, deriving a set of invariant features that are based on geometrical moments of the images. Fully connected neural networks of various sizes and number of hidden layers were considered, using the same set of image data, as above, for invariant classification of the computed feature set. However, the learning and generalization ability of the networks was rather poor, being able to classify correctly less than 80 percent of the image data.

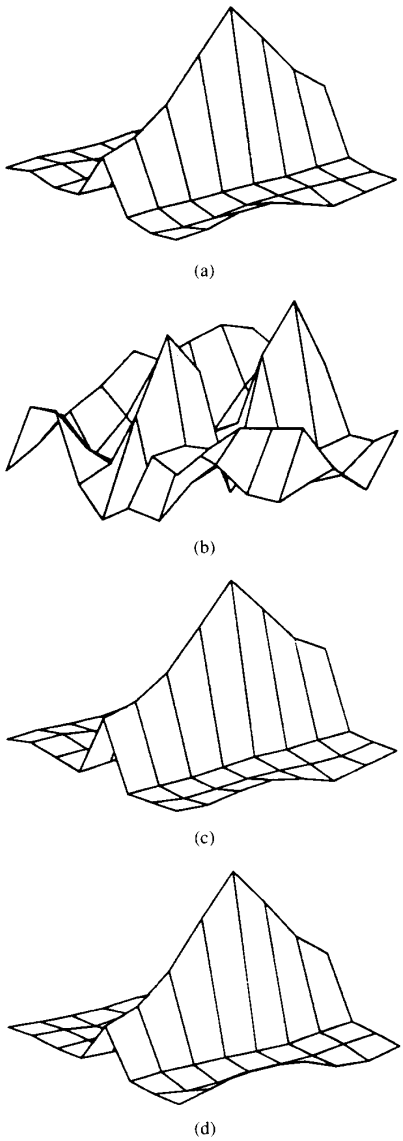


Fig. 16. Content of class (3°, 1°) in each of the representations shown in Fig. 15.

VI. Conclusions

A new technique for invariant image classification has been introduced in this paper. Triple-correlation-based neural net-

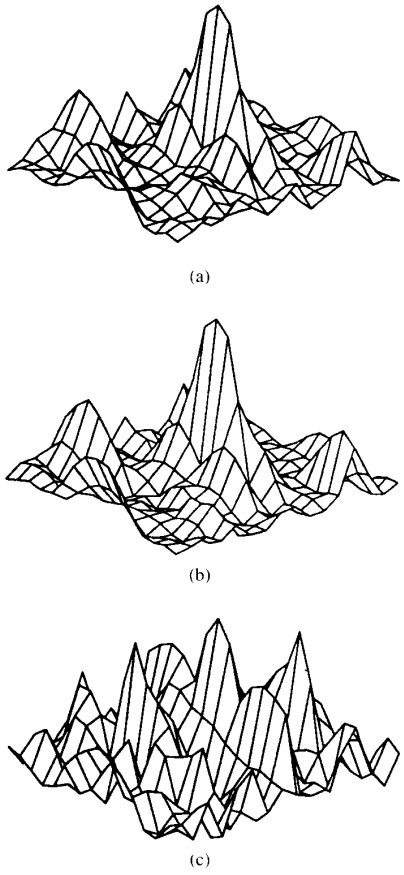


Fig. 17. Invariant representations using the Fourier amplitude and log-polar image transformation, (a) of the image in Fig. 10(a); (b) of the same image, rotated by 45°; (c) of the image in Fig. 10(b).

works were defined and proposed for this purpose. Appropriate clustering of third-order correlations of an image have been used to provide a representation that is invariant with respect to translation, rotation, scale and insensitive to additive noise. An efficient implementation scheme has been then proposed, derived on a two-angle (θ_1, θ_2) image plane, that reduces the problem size, providing also robustness to distortions of the input image.

A set of representations, of gradually reducing complexity, has been derived in the form of 2-D images and used as a direct input to appropriately selected neural network architectures. It has also been shown that third-order neural networks are a specific category of triple-correlation-based neural networks, being, therefore, amenable to the efficient implementation proposed in this paper.

Synthetic and real image data were used to illustrate various aspects of the proposed invariant classification scheme. Based on these results it can be concluded that triple-correlation-based neural network classifiers are an effective and powerful classification scheme. Further study of the implementation of the proposed technique, both in software and hardware, is currently under investigation.

APPENDIX A

Proof of Proposition 4

Let us first observe that classes $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ and $C(\boldsymbol{T}_{\alpha,\theta}\boldsymbol{\tau}_1, \boldsymbol{T}_{\alpha,\theta}\boldsymbol{\tau}_2)$ are equivalent for any $\boldsymbol{T}_{\alpha,\theta}$ of the form (15), in the sense that the latter is a circularly shifted by $(\log\alpha, \theta)$ version of the first. Hence, fixing $\boldsymbol{\tau}_1$ to τ_0 and letting $\boldsymbol{\tau}_2$ span S' generates all essentially distinct classes. We next use triple-correlation symmetries [c.f. (5)] to restrict $\boldsymbol{\tau}_2$. Due to (5) and in view of (14) and the definition of $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ it can be easily checked that,

$$C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2) \equiv C(\boldsymbol{\tau}_1 - \boldsymbol{\tau}_2, \boldsymbol{\tau}_2) \equiv C(\boldsymbol{\tau}_2 - \boldsymbol{\tau}_1, -\boldsymbol{\tau}_2) \equiv$$

$$C(\boldsymbol{\tau}_2, \boldsymbol{\tau}_1) \equiv C(-\boldsymbol{\tau}_1, \boldsymbol{\tau}_2 - \boldsymbol{\tau}_1) \equiv C(-\boldsymbol{\tau}_2, \boldsymbol{\tau}_1 - \boldsymbol{\tau}_2). \quad (42)$$

in the sense that they contain the same triple-correlation lags at the same locations. For example,

$$x_3(\boldsymbol{T}_{\beta,\phi}\boldsymbol{\tau}_1, \boldsymbol{T}_{\beta,\phi}\boldsymbol{\tau}_2) = x_3(\boldsymbol{T}_{\beta,\phi}(\boldsymbol{\tau}_1 - \boldsymbol{\tau}_2), \boldsymbol{T}_{\beta,\phi}(-\boldsymbol{\tau}_2)), \quad (43)$$

i.e., at location $(\log\beta, \phi)$, $C(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)$ and $C(\boldsymbol{\tau}_1 - \boldsymbol{\tau}_2, -\boldsymbol{\tau}_2)$ have identical values for any $\beta > 0$ and $\phi \in [-\pi, \pi]$. Based on (43) we use simple matrix multiplication arguments to show that for $\boldsymbol{\tau}_1 = \tau_0$ and $\boldsymbol{\tau}_2$ in the fourth quadrant, there exists some $\boldsymbol{\tau}_2'$ in the first quadrant such that

$$C(\tau_0, \boldsymbol{\tau}_2) = C(\tau_0, \boldsymbol{\tau}_2'), \quad (44)$$

within a shift transformation. Indeed, for $\boldsymbol{\tau}_2$ in the fourth quadrant, there exists θ with $0 < \theta < \pi/2$ such that $\boldsymbol{T}_{\alpha,\theta}\boldsymbol{\tau}_2 = \tau_0$, because,

$$\begin{aligned} C(\tau_0, \boldsymbol{\tau}_2) &= C(\boldsymbol{T}_{\alpha,\theta}\tau_0, \boldsymbol{T}_{\alpha,\theta}\boldsymbol{\tau}_2) \\ &= C(\boldsymbol{T}_{\alpha,\theta}\boldsymbol{\tau}_2, \boldsymbol{T}_{\alpha,\theta}\tau_0) \\ &\equiv C(\tau_0, \boldsymbol{\tau}_2'). \end{aligned} \quad (45)$$

with $\boldsymbol{\tau}_2' = \boldsymbol{T}_{\alpha,\theta}\tau_0$ in the first quadrant. In the second equality in (45) we interchanged the indices of $C(.,.)$ using the first and fourth terms of (42).

In the same manner and using other symmetries in (42) we can prove that the second and third quadrants are also mapped into the first one.

We proceed now to prove that any $\boldsymbol{\tau}_2$ in the first quadrant out of the zone $S_0 = [1, 0] \times [0, \infty)$, generates a class $C(\tau_0, \boldsymbol{\tau}_2)$ equivalent to some $C(\tau_0, \boldsymbol{\tau}_2')$ in S_0. Notice first that $\boldsymbol{\tau}_2 = \boldsymbol{T}_{\alpha,\theta}\tau_0$ and $\boldsymbol{\tau}_2 - \tau_0 = \boldsymbol{T}_{\rho,\omega}\tau_0$, where,

$$\begin{aligned} 0 \le \theta \le \omega \le \frac{\pi}{2}, \\ \alpha > 1 \text{and} \\ \rho \le \alpha. \end{aligned} \quad (46)$$

Also $-\boldsymbol{\tau}_2 = T_{\alpha,\theta+\pi}\tau_0$ and $\tau_0 - \boldsymbol{\tau}_2 = T_{\rho,\omega+\pi}\tau_0$. Consequently, using (42),

$$\begin{aligned} C(\tau_0, \boldsymbol{\tau}_2) &= C(-\boldsymbol{\tau}_2, \tau_0 - \boldsymbol{\tau}_2) \\ &= C(T_{\alpha,\theta+\pi}\tau_0, T_{\rho,\omega+\pi}\tau_0) \\ &= C\left(\tau_0, T_{\frac{\rho}{\alpha},\omega-\theta}\tau_0\right) \\ &= C(\tau_0, \boldsymbol{\tau}_2'), \end{aligned} \quad (47)$$

where $\tau_2' = T_{\frac{\varrho}{\alpha},\omega-\theta}\tau_0$ due to (46) lies in the zone S_0. This completes the proof of Proposition 4.

ACKNOWLEDGMENT

We wish to thank SIEMENS AG, Munich, and especially Dr. Peter Mengel, for providing us with the solder joint image data, that have also been used in the ANNIE project. We also wish to thank the anonymous reviewers for their constructive comments.

REFERENCES

[1] E. Bienestock and C. A. Von der Malsburg, "A neural network for the retrieval of superimposed connection patterns," *Europhysics Letters*, vol. 3, pp. 1243–1249, 1987.
[2] V. Chandran, and S. Elgar, "Shape discrimination using invariant features defined from higher-order spectra," in *Proc. of Intl. Conf. on Acoustics Speech and Signal Processing*, vol. 5, Toronto, Canada, May 14–18, 1991, pp. 3105–3109.
[3] V. Chandran, and S. Elgar, "Position, rotation, and scale invariant recognition of images using higher-order spectra," in *Proc. of Intl. Conf. on Acoustics Speech and Signal Processing*, vol. 5, San Fransisco, CA, March 1992, pp. 213–216.
[4] Y. Chauvin, "A back-propagation algorithm with optimal use of the hidden units," in *Advances in Neural Information Processing Systems I*, D. Tourentry, Ed. Palo Alto, CA: Morgan Kaufman, 1989.
[5] A. V. Dandawate and G. B. Giannakis, "Ergodicity and asymptotic normality for cumulant estimators of nonstationary signals," *Proc. of 25th Conf. on Info. Sciences and Systems*, The Johns Hopkins Univ., Baltimore, March 1991, pp. 976–983.
[6] J. Denker *et al.*, "Handwritten digit recognition: Applications of network chips and automatic learning," *IEEE Communications Magazine*, pp. 41–46, 1989.
[7] R. Duda and P. Hart, *Pattern Classification and Scene Analysis*. New York: John Wiley and Sons, 1973.
[8] K. Fukushima, S. Miyake, and T. Ito, "Neocognitron: A neural model for a mechanism of visual pattern recognition," *IEEE Trans. on Sys. Man and Cyb.*, vol. 13, no. 5, pp. 826–834, 1983.
[9] K. Fukushima, "Neocognitron: A hierarchical neural network capable of visual pattern recognition," *Neural Networks*, vol. 1, pp. 119–130, 1988.
[10] G. B. Giannakis, and M. K. Tsatsanis, "Signal detection and classification using matched filtering and higher-order statistics," *IEEE Trans. on Acoustics Speech and Signal Processing*, pp. 1284–1296, 1990.
[11] G. B. Giannakis, and M. K. Tsatsanis, "A unifying maximum-likelihood view of cumulant and polyspectral measures for non-Gaussian signal classification and estimation," *IEEE Trans. on Information Theory*, vol. 38, pp. 386–406, 1992.
[12] G. L. Giles and T. Maxwell, "Learning, invariance and generalization in higher-order neural networks," *Applied Optics*, vol. 26, 1987.
[13] G. L. Giles, R. D. Griffin, and T. Maxwell, "Encoding geometric invariances in higher-order neural networks," *Neural Information Processing Systems*, American Institute of Physics Conference Proceedings, 1988.
[14] S. J. Hanson, "Comparing biases for minimal network construction with back-propagation," in *Advances in Neural Information Processing Systems I*, D. Tourentry, Ed. Palo Alto, CA: Morgan Kaufman, 1989.
[15] M. H. Hayes, "The reconstruction of a multidimensional sequence from the phase or magnitude of its Fourier transform," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 30, no. 2, pp. 140–154, 1982.
[16] R. Jacobs, "Increased rates of convergence through learning rate adaptation," *Neural Networks*, vol. 1, no. 4, 1988.
[17] A. Khotanzad and J. H. Lu, "Classification of invariant image representations using a neural network," *IEEE Trans. on ASSP*, vol. 38, pp. 1028–1038, 1990.
[18] S. Kollias and D. Anastassiou, "An adaptive least squares algorithm for the efficient training of artificial neural networks," *IEEE Trans. on Circuits and Systems*, vol. 36, pp. 1092–1101, 1989.
[19] S. Kollias, A. Stafylopatis, and A. Tirakis, "Performance of higher order neural networks in invariant recognition," in *Neural Networks: Advances and Applications*. North Holland, 1991, pp. 79–108.
[20] Y. LeCun, "Generalization and network design strategies," in *Connectionism in Perspective*. Switzerland: North Holland, pp. 143–155, 1989.
[21] Y. LeCun, "A theoretical framework for backpropagation," in *Proceedings of the 1988 Connectionist Models Summer School*, CMU, Pittsburg, 1989, San Mateo, CA: Morgan Kaufman.
[22] C. Von der Malsburg, "Pattern recognition by labelled graph matching," *Neural Networks*, vol. 1, pp. 141–148, 1988.
[23] J. M. Mendel, "Tutorial in higher-order statistics (spectra) in signal processing and system theory: Theoretical results and some applications," *Proc. IEEE*, vol. 79, pp. 278–305, 1991.
[24] A. V. Oppenheim, and J. L. Lim, "Importance of phase in signals," *Proc. IEEE*, vol. 69, pp. 529–541, 1981.
[25] Y. H. Pao, *Adaptive Pattern Recognition and Neural Networks*. Reading, MA: Addison-Welsey, 1989.
[26] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*. Third Edition, McGraw-Hill International Editions, Electrical & Electronic Engineering Series, 1991, pp. 212–213.
[27] M. B. Reid, L. Spirkovska, and E. Ochoa, "Rapid training of higher-order neural networks for invariant pattern recognition," in *Proc. Joint Int. Conf. on Neural Networks*, Washington, D.C., June 1989.
[28] M. B. Reid, L. Spirkovska, and E. Ochoa, "Simultaneous position, scale, and rotation invariant pattern classification using third-order neural networks," *The International Journal of Neural Networks*, vol. 1, no. 3, July 1989.
[29] D. Rumelhart, and J. McClelland, Eds., *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. Cambridge, MA: MIT Press, vol. 1, 1986.
[30] M. K. Tsatsanis and G. B. Giannakis, "Object and texture classification using mathced filtering and higher-order statistics," *Proc. of 6th Workshop on Multidimensional Signal Processing*, Monterey, CA, September 1989, pp. 32–33.
[31] M. K. Tsatsanis, and G. B. Giannakis, "Translation, rotation and scaling invariant object and texture classification using polyspectra," in *Proc. of Soc. of Photo-Opt. Instr. Engr., Advanced Signal Processing Alg., Arch. and Implem.*, San Diego, CA, July 1990, vol. 1348, pp. 103–115.
[32] M. K. Tsatsanis, and G. B. Giannakis, "Object and texture classification using higher-order statistics," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 733–750, 1992.
[33] B. Widrow and R. Winter, "Neural nets for adaptive filtering and adaptive pattern recognition," *IEEE Computer*, vol. 21, no. 3, pp. 25–39, 1988.
[34] B. Yegnanarayana, D. K. Saikia, and T. R. Krishnan, "Significance of group delay functions in signal reconstruction from spectral magnitude or phase," *IEEE Trans. on ASSP*, vol. 32, no. 3, pp. 610–622, 1984.
[35] ANNIE, "Applications of Neural Networks to Industrial Pattern Recognition Problems," Esprit Project 2092 Handbook, 1991.

Anastasios N. Delopoulos (S'88–M'92) was born in Athens, Greece, in 1964. He graduated from the Department of Electrical Engineering, the National Technical University of Athens (NTUA), in 1987, and received the M.Sc. degree in electrical engineering from the University of Virginia, U.S.A., in 1990, and the Ph.D. degree in electrical and computer engineering from the NTUA in 1993.

His current research interests lie in the areas of system identification, video coding, massively parallel architectures, and biomedical engineering.

Dr. Delopoulos is a member of the Technical Chamber of Greece and a member of the IEEE Signal Processing Society.

Andreas D. Tirakis was born in Athens, Greece, in 1965. He received the Diploma degree in electrical engineering from the National Technical University of Athens (NTUA) in 1989. He is currently working toward the Ph.D. degree in the Computer Science Division of NTUA.

His research interests lie in the areas of digital image processing, computer vision, and artificial neural networks.

Mr. Tirakis is a member of the Technical Chamber of Greece.

Stefanos D. Kollias (S'81–M'84) was born in Athens, Greece, in 1956. He received the Diploma degree in electrical engineering from the National Technical University of Athens (NTUA) in 1979, the M.Sc. degree in communication engineering from the University of Manchester Institute of Science and Technology, U.K., in 1980, and the Ph.D. degree in signal processing from the Computer Science Division of NTUA in 1984.

Since 1984, he has been with the Computer Science Division of NTUA, where he is currently an Associate Professor. In 1987–88 he was a visiting research scientist in the Electrical Engineering Department of Columbia University, New York, on leave from NTUA. His research interests include digital signal and image processing and analysis, artificial neural networks, and parallel architectures.