Generalized Information Potential Criterion for Adaptive System Training

Deniz Erdogmus, *Student Member, IEEE,* and Jose C. Principe, *Fellow, IEEE*

***Abstract*—We have recently proposed the quadratic Renyi's error entropy as an alternative cost function for supervised adaptive system training. An entropy criterion instructs the minimization of the average information content of the error signal rather than merely trying to minimize its energy. In this paper, we propose a generalization of the error entropy criterion that enables the use of any order of Renyi's entropy and any suitable kernel function in density estimation. It is shown that the proposed entropy estimator preserves the global minimum of actual entropy. The equivalence between global optimization by convolution smoothing and the convolution by the kernel in Parzen windowing is also discussed. Simulation results are presented for time-series prediction and classification where experimental demonstration of all the theoretical concepts is presented.**

***Index Terms*—Minimum error entropy, Parzen windowing, Renyi's entropy, supervised training.**

I. Introduction

THE mean-square error (MSE) has been the workhorse of adaptive systems research due to the various analytical and computational simplicities it brings and the contentment of minimizing error energy in the framework of linear signal processing. In a statistical learning sense, especially for nonlinear signal processing, a more appropriate approach would be to constrain directly the information content of signals rather than simply their energy, if the designer seeks to achieve the best performance in terms of information filtering [1]–[3].

A measure of information is the entropy, defined as the average information [4]. Entropy was first defined and proved useful by Shannon in the context of communication systems [5]. Under the Gaussian assumption, Shannon's entropy and mutual information are still mathematically tractable. Many entropy definitions followed Shannon's from which we single out Kolmogorov's entropy very useful in statistics [6]. A shortcoming of these entropy definitions is the lack of computationally simple and efficient nonparametric estimators for non-Gaussian random processes, particularly in high-dimensional spaces. For example, Viola uses the simple sample mean to estimate mutual information [7] and concludes that this estimator breaks down in high-dimensional spaces. In the blind source separation (BSS) context, Shannon's entropy is estimated from the samples using truncated polynomial expansions of the densities under consideration [8], [9]. These reasons, along with the property that the convolution of two Gaussian functions is also Gaussian, were the motivating factors to investigate Parzen windowing [10] with Renyi's quadratic entropy [11]. These authors have performed a significant amount of research on the performance and applicability of this approach to various problems including BSS [11]–[13], feature extraction, data reduction, feature ranking, and data fusion [11], [14]. We do not know of any other work that uses kernel estimation to build cost functions in learning. Vapnik uses kernels to transform data into high-dimensional spaces in his support vector machines [15] and kernel estimation in the context of signal/information processing has been applied as a measure of similarity between chaotic time series [16]; it is also the basis to estimate attractor dimension [17]. We have recently demonstrated the superiority of the entropy criterion over the MSE in chaotic time series prediction with time-delay neural networks (TDNNs) [18]. The results indicated that the entropy-trained TDNN achieved a better fit to the density of the desired samples, because minimizing error entropy minimizes a Riemannian distance between the system output and the desired output densities [19]. In a follow-up study, it was shown that the entropy estimator also outperforms other methods for BSS [12].

Renyi's entropy is a parametric function family [20]. In this paper, we provide an extension to the previously suggested quadratic $(\alpha = 2)$ Renyi's entropy estimator, which makes possible the choice of any entropy order and kernel function. It can be shown using L'Hôpital's rule that Shannon's entropy is the limiting value of the Renyi's entropy when α approaches one [20]. This property motivates the usage of Renyi's entropy and Parzen window estimator, with an α value close to one as a computationally efficient alternative to estimate Shannon's entropy.

Parzen windowing is a consistent estimator, yet it has some problems; it is a biased density estimator where the density's expected value is equal to the convolution of the actual density that produced the samples, and the kernel function [10]. However, we will show that there is a way to exploit this smoothing property and use it to our advantage as a means of avoiding local optima in the training process and thus achieving global optimization.

The organization of this paper is as follows. First, we derive the estimator for Renyi's entropy in Section II and investigate some of its mathematical properties. Next, we define the order-α information potential and information forces, study their relationship with their quadratic counterparts, and demonstrate their role in the training process in Section III. This investigation is followed by the presentation of the supervised steepest descent training algorithm for adaptive systems using the entropy as the performance measure. In Section V, we demonstrate the link

Manuscript received February 21, 2000; revised August 23, 2001. This work was supported in part by NSF under Grant ECS-9900394.

The authors are with the Computational Neuroengineering Laboratory, University of Florida, Gainesville, FL 32611 USA.

Publisher Item Identifier S 1045-9227(02)04433-8.

between our estimation method and the convolution smoothing method of global optimization. Finally, we present experimental results from prediction and classification in Section VI, followed by a discussion and conclusion section.

II. Estimating Renyi's Entropy With Parzen Windowing

Renyi's entropy with parameter α for a random variable e with probability density function (pdf) $f_e(\cdot)$ is defined as (we will drop the $\pm\infty$ limits from the equations from now on)

$$H_\alpha(e) = \frac{1}{1-\alpha}\log\int_{-\infty}^{\infty} f_e^\alpha(e)\,de. \tag{1}$$

Renyi's entropy shares the same extreme points of Shannon's definition for all values of α, i.e., its minimum value occurs when $f_e(\cdot)$ is a Dirac-δ function and the maximum occurs when the pdf is uniform. Since in realistic problems the analytical expression for the pdf is hardly ever known, a productive way is to estimate nonparametrically the density from the samples. That is where Parzen windowing comes into play. The pdf estimate of a random variable e for which the samples $\{e_1,\ldots,e_N\}$ are given is obtained using the kernel function $\kappa_\sigma(\cdot)$, whose size is specified by the parameter σ, with the following expression:

$$\hat{f}_e(e) = \frac{1}{N}\sum_{i=1}^{N}\kappa_\sigma(e-e_i). \tag{2}$$

In [11], Gaussian kernels were specifically utilized and α was restricted to two, yielding

$$\begin{aligned}\hat{H}_2(e) &= -\log\hat{V}(e)\\ \hat{V}(e) &= \frac{1}{N^2}\sum_j\sum_i G_{\sigma\sqrt{2}}(e_j-e_i)\end{aligned} \tag{3}$$

where σ is the standard deviation of the Gaussian kernel used in Parzen windowing and $V(e)$ the argument of the log in Renyi's entropy is called the *information potential* [11]. We will now generalize these choices. We first observe that the integral of $f_e^\alpha(e)$ in (1) is the expected value of $f_e^{\alpha-1}(e)$ and then substitute the sample mean for the expected value operator yielding

$$\begin{aligned}H_\alpha(e) &= \frac{1}{1-\alpha}\log E\left[f_e^{\alpha-1}(e)\right]\\ &\approx \frac{1}{1-\alpha}\log\left[\frac{1}{N}\sum_j f_e^{\alpha-1}(e_j)\right].\end{aligned} \tag{4}$$

Then we replace the actual pdf with the Parzen window estimator to obtain our estimator for Renyi's entropy of order α

$$H_\alpha(e) = \frac{1}{1-\alpha}\log\left[\frac{1}{N^\alpha}\sum_j\left(\sum_i\kappa_\sigma(e_j-e_i)\right)^{\alpha-1}\right]. \tag{5}$$

The information potential is still defined as the argument of the log (see [11]) and will come in handy when we specify a value for α. Minimizing the entropy is equivalent to maximizing the information potential for $\alpha>1$, or minimizing the information potential for $\alpha<1$ since the log is a monotonous function. This means that in entropy manipulation we can utilize simply the information potential as our cost function.

The minimum value of the entropy will be achieved for a δ-distributed random variable. The question is, when we have the samples of this random variable as $e_1=\cdots=e_N=0$ and we use the nonparametric estimator in (5), is this still a minimum of the estimated entropy? The following theorem addresses this question.

Lemma 1: If a symmetric, continuous and differentiable, and unimodal kernel with a peak at the origin is used, then the nonparametric entropy estimator in (5) has a continuous and differentiable local minimum when all the samples are identically equal to each other.

Proof: See Appendix A. □

Theorem 1: If a symmetric, continuous and differentiable, and unimodal kernel is used, then the smooth local minimum described in Lemma 1, which occurs when all samples are equal, is the global minimum of the entropy estimator in (5).

Proof: See Appendix A. □

This result is significant because it proves that we have an estimator that preserves the global minimum under certain constraints imposed on the structure of the estimator, namely, the choice of the kernel function. In addition, the result also yields the profile of the cost function in the neighborhood of this global minimum by identifying the corresponding local eigenstructure. It is possible to design a suitable kernel function and, hence, an estimator, by analyzing this structure.

Up to this point, we discussed some properties of the entropy estimator in (5). Now, we show how minimizing error entropy achieves the best statistical learning solution in terms of pdf matching. This theoretical result based on Csiszar's distance [21] is also confirmed by simulation results that will be presented later.

Theorem 2: Minimizing Renyi's error entropy minimizes a Csiszar distance between the joint pdfs of input-desired signals and the input–output signals. In the special case of Shannon's entropy, this Csiszar distance measure reduces to the Kullback-Leibler divergence.

Proof: See Appendix A. □

III. Information Potential Field and Information Forces

The use of kernels and the resulting entropy formulation introduces an interesting interpretation of the training process. The name information potential is not arbitrary. In fact, the concept of information potential fields generated by samples seen as information particles and the forces they exert on each other were defined and investigated for the quadratic entropy with Gaussian kernels in [11]. It is now possible to define the order-α information potentials and information forces among samples and, thus, define the relationship between these new and the old quadratic quantities. We define the information potential estimator as

$$\hat{V}_{\alpha,\sigma}(e) = \frac{1}{N^\alpha}\sum_j\left(\sum_i\kappa_\sigma(e_j-e_i)\right)^{\alpha-1} \tag{6}$$

where the n-dimensional size-σ kernel can be written in terms of the unit-size kernel according to (7) (notice that in the one-dimensional case, the standard deviation acts as a natural scaling factor for the Gaussian kernel)

$$\kappa_\sigma(x) = \frac{1}{\sigma^n}\kappa(x/\sigma). \tag{7}$$

It is important to remark that (6) reduces to (3) for $\alpha = 2$ and when Gaussian kernels of twice the variance are used in the estimation. This shows the interesting fact that, in the estimation of Renyi's entropy, the sample mean approximation in (3) can be *exactly* compensated by an appropriate choice of the Parzen window function [22].

We can write the potential associated with an information particle (sample) e_j from the above expression, since the total information potential energy is the sum of individual energies of the particles

$$\hat{V}_{\alpha,\sigma}(e_j) = \frac{1}{N^\alpha}\left(\sum_i \kappa_\sigma(e_j - e_i)\right)^{\alpha-1}. \tag{8}$$

From this, we can compute the total information force $F_\alpha(e_j)$ acting on e_j by making the physical analogy with forces in potential fields as

$$\begin{aligned} F_\alpha(e_j) &= \frac{\partial \hat{V}_\alpha(e_j)}{\partial e_j} \\ &= \frac{(\alpha-1)}{N^\alpha}\left(\sum_i \kappa_\sigma(e_j - e_i)\right)^{\alpha-2} \\ &\quad \times \left(\sum_{i\neq j} \kappa'_\sigma(e_j - e_i)\right). \end{aligned} \tag{9}$$

A more informative equivalent expression for the information force on particle e_j is

$$F_\alpha(e_j) = (\alpha-1)\hat{f}_e^{\alpha-2}(e_j)F_2(e_j) \tag{10}$$

where the quadratic force is defined as

$$F_2(e_j) = \frac{1}{N^2}\left(\sum_{i\neq j} \kappa'_\sigma(e_j - e_i)\right) \tag{11}$$

which makes perfectly clear the relationship of the α-force and the quadratic force.

This quadratic force expression reduces to *exactly* the same definition in [11], when Gaussian kernels of twice the variance are employed. Now combining (10) and (11), we can define the force exerted on a sample e_i by another particular sample e_i as

$$F_\alpha(e_j; e_i) = (\alpha-1)\hat{f}_e^{\alpha-2}(e_j)F_2(e_j; e_i). \tag{12}$$

Having completed the formulation of information forces and having established the link between the order-α force and the quadratic force, we can now interpret the expressions. Basically, the quadratic force can be regarded as the foundation for all other information forces. Forces of any order can be represented as a scaled version of the quadratic force, where the scaling factor is a power of the value of the probability density of the particle that the force acts upon. For $\alpha > 2$, the force on a particle increases with increased probability density, while it decreases for $\alpha < 2$.

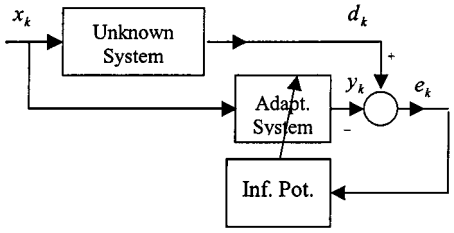

Fig. 1. Adaptive system training using information potential criterion.

In this section, we have established a basis for minimum error entropy training. We have analyzed the relations of the general information potential and force expressions with their quadratic counterparts. These links enabled us to understand how the choice of the entropy parameter α affects the information forces, hence the adaptation process. We have learned that the information force acting on an information particle is scaled up by a power of its probability density estimate, therefore, selecting an α greater than two will result in higher forces acting on samples in more concentrated regions of the data space, whereas choosing a smaller α will result in higher forces acting on samples that are in less populated regions. In the context of BSS, it was made clear that the use of $\alpha > 2$ for super-Gaussian distributed samples was preferable, whereas $\alpha < 2$ was suggested for sub-Gaussian signals [22]. Other entropy orders, however, will also provide satisfactory results unlike other methods based on the sign of the kurtosis [23]. If the residual error is anticipated to be Gaussian, or if the designer does not have any *a priori* information about the sign of the source kurtosis, the quadratic entropy is the best choice.

IV. Gradient Training Algorithm for Information Potential

Let us assume that the design goal is to adapt the parameters of a linear or nonlinear parametric mapper $y = g(x, w)$ in a function approximation framework, as schematically shown in Fig. 1. We define the error as the difference between the desired output and the output of the mapper to a corresponding input. Instead of the MSE, we will be using an information theoretic learning criterion based on the information potential in (6).

Our aim is to minimize the error entropy since we know that doing so, we would also achieve the best statistical fit to the joint density between the input signal and the desired output. We have demonstrated in Theorem 2 that minimizing error entropy achieves this goal. On the other hand, noticing that the logarithm is a monotonic function, it is much easier to equivalently minimize or maximize the information potential for the cases $\alpha < 1$ and $\alpha > 1$, respectively. This simplification brings about significant computational savings by eliminating the need to evaluate the information potential at every step of the gradient update algorithm.

The information forces are encountered when training an adaptive system with weight vector w with the information potential criterion and using a gradient-based method. We

adapt the parameters of the mapper by injecting the information force for sample e_j as the external error. The gradient of the information potential of the error with respect to the weights consists of products of the information force acting on an error sample, and the sensitivity of the architecture at that error value

$$\frac{\partial \hat{V}_\alpha(e)}{\partial w} = \sum_j \frac{\partial \hat{V}_\alpha(e_j)}{\partial e_j}\frac{\partial e_j}{\partial w} = \sum_j F_\alpha(e_j) S_w(e_j). \quad (13)$$

On the other hand, the gradient of the information potential with respect to the weights of the adaptive system is explicitly given by

$$\frac{\partial \hat{V}_\alpha}{\partial w} = \frac{(\alpha-1)}{N^\alpha}\sum_j \left(\sum_i \kappa_\sigma(e_j - e_i)\right)^{\alpha-2} \times \left[\sum_i \kappa'_\sigma(e_j - e_i)\left(\frac{\partial y_i}{\partial w} - \frac{\partial y_j}{\partial w}\right)\right] \quad (14)$$

where the desired response, system output, and error samples are defined as d_k, y_k, and $e_k = d_k - y_k$. The instantaneous gradient of the system output with respect to the weights can be calculated using efficient methods depending on the type of adaptive system used; for example for a finite impulse response (FIR) filter, this gradient will simply be the corresponding input vector as in the least-mean square (LMS) algorithm [24], and for a multilayer perceptron (MLP) it can be computed by backpropagation [25].

It is of theoretical interest to investigate the relation between the gradient of the order-α information potential and the quadratic information potential. The expression in (13) can be rearranged to yield

$$\frac{\partial \hat{V}_\alpha(e)}{\partial w} = (\alpha - 1)\sum_j \hat{f}_e^{\alpha-2}(e_j)\frac{\partial \hat{V}_2(e_j)}{\partial w} \quad (15)$$

whereas the total gradient of quadratic information potential is simply the sum of the gradients of each particle e_j, that is, it corresponds to an equal mixing of the individual gradients generated by each particle (with a scale of N). However, in the order-α case, the total gradient is a weighted mixture of the individual gradients created by each particle where the mixture coefficients are the powers of the pdf estimate of the corresponding particle. This property directly translates from what was observed for the information forces.

Gradient adaptation is not the only search possibility, but it is preferred in many training paradigms due to its simplicity and efficient convergence [24]. Yet, there are many other alternative optimization approaches that may be used, global or otherwise [26], [27].

V. Kernel Size and Global Optimization by Functional Smoothing

Thus far, we have derived a performance criterion and the corresponding gradient-based supervised learning algorithm for adaptive system training. The method can be used to train systems ranging from linear FIRs to nonlinear neural-networks systems. Now it is time to ask the question, "what are the mathematical properties of the combination of this criterion, with the training algorithm?" In pursuit of an answer to this question, we investigate the effect of the kernel size on the criterion. The kernel size σ is a very important parameter if efficiently exploited. Since Parzen windowing is a biased estimator of the pdf where the tradeoff is between low bias and low variance, one must choose a suitable value small enough to minimize the bias at an acceptable variance level. The kernel size cannot be chosen to be zero because this would mean placing δ-functions on the samples and although this would make the bias zero, it would blow up the variance of the pdf estimation.

Once a suitable value is set, training can be carried out using that fixed kernel size value. Our experience shows that this value is not critical in the final performance as long as it is not extremely small or extremely large. For example, the kernel size can be set to a value so that each kernel will cover, say, ten samples on average over the foreseen dynamic range of the error. This has been the way we have applied the method until now. However, due to the nonlinear nature of most function approximation problems, some local optima may exist. It turns out that the kernel size may be effectively used to avoid these local optima. Consider the following relation that identifies the link between the order-α information potential of a given set of error samples for an arbitrary kernel size and for the unit size kernel

$$\hat{V}_{\alpha,\sigma}(e) = \frac{1}{N^\alpha}\sum_j \left(\sum_i \frac{1}{\sigma^n}\kappa\left(\frac{e_j - e_i}{\sigma}\right)\right)^{\alpha-1} = \frac{1}{\sigma^{n(\alpha-1)}}\hat{V}_{\alpha,1}(e/\sigma). \quad (16)$$

Notice that the change in kernel size causes dilation in the e-space. Therefore, all the points, including all local extremes move radially from the origin when σ is increased. The only point that maintains its position is the origin. From this, we conclude that if the span of the function approximator that is being used covers the function being approximated (i.e., the error is approximately zero), then the location of the global solution is independent of the kernel size. Also, if the function approximator used is a contractive mapping, which is the case in feedforward neural networks for example, then the dilation in the e-space is followed by dilation in the weight-space, hence, the volume of attraction of the global optimum is increased. This observation lead us to propose a *global optimization* procedure for the training process based on gradient descent and annealing kernel sizes. If one starts with a large kernel size and during the adaptation gradually and slowly decrease it toward the predetermined suitable value, the local solutions, which would have trapped the training for those same initial conditions when the W-space was not dilated, will be avoided. Hence, we obtain a global optimization result still using a gradient descent approach!

In addition to the dilation property in the finite sample case, there is another interesting property that the Parzen windowing brings about. In the literature of global optimization, a well-known theoretical result uses convolution of the cost function with a suitable smooth function to eliminate local optima and gradually decrease the effect of the convolution to achieve global optimization [28]. Convolution smoothing was proven effective in adaptation of infinite impulse response (IIR) filters [29]. The

global convergence theorem for convolution smoothing states that the following optimization problems are equivalent:

$$\min_{x\in D\subset R^n} g(x) = g(x^*) = \min_{x\in D\subset R^n} \hat{g}_\beta(x),\ \beta \to 0 \tag{17}$$

where the smoothened functional is defined as

$$\hat{g}_\beta(x) = g(x)^* h_\beta(x) \tag{18}$$

and, thus, both result in the global optimal point x^* [28]. There are conditions that the smoothing functional $h_\beta(x)$ has to satisfy

$$\begin{aligned} &1)\quad h_\beta(x) = (1/\beta^n)h(x/\beta)\\ &2)\quad \lim_{\beta\to 0} h_\beta(x) = \delta(x)\\ &3)\quad \lim_{\beta\to 0} \hat{g}_\beta(x) = g(x)\\ &4)\quad h_\beta(x)\ \text{is a pdf.} \end{aligned} \tag{19}$$

Condition 3 guarantees that both $g(x)$ and $h_\beta(x)$ are well-behaved functions. Condition 4 gives the problem a stochastic optimization flavor [28]. For our purposes, this strict condition is not a problem since even if the convolving function does not integrate to one then the same convolution smoothing effect will be observed, except there will be a scale factor that multiplies the smoothed functional. The most important constraints on the smoothing function are Conditions 1) and 2).

In the supervised training process, if we could obtain an analytical expression for the pdf of the error as a function of the weights, then we would optimize the actual information potential given by

$$V_\alpha(w) = \int f_e^\alpha(e;w)\,de. \tag{20}$$

However, we are using Parzen windowing to estimate $f_e(e;w)$ from the samples. Since Parzen windowing is a consistent estimator, as $N\to\infty$, the estimated pdf converges to the actual pdf convolved with the kernel function that is used, which also happens in the mean

$$\hat{V}_{\alpha,\sigma}(w) = \int [f_e(e;w) *_e \kappa_\sigma(e)]^\alpha\, de \tag{21}$$

where $*_e$ denotes a convolution with respect to the variable e. When we equate this to the convolution smoothed information potential, i.e., (20) and (21), we get

$$\hat{V}_{\alpha,\sigma}(w) = V_\alpha(w)^* h_\beta(w). \tag{22}$$

Although it is not easy to solve for the corresponding smoothing function from this equation, we may be able to show that the solution still satisfies the required conditions. At this point, the authors can show that functionals $h_\beta(w)$ solutions to (22), satisfy Conditions 2)–4) of (19). Furthermore, the property of dilation in the e-space presented in (16) hints toward the validity of Condition 1). However, it was not possible to verify that the first condition is satisfied in general for any mapper, nor it was possible to set forth the conditions under which this occurs. Therefore, we propose the existence of such a smoothing functional corresponding to each kernel function choice as a conjecture.

Conjecture 1: Given a specific choice of the kernel function $\kappa_\sigma(\cdot)$, there exists a corresponding smoothing functional $h_\beta(\cdot)$, which is a solution of (22) and satisfies Conditions 1)–4). (See Appendix B for the solution.)

In this section, we have analyzed the effect of the kernel size on the criterion and the search algorithm. We have shown the link between our information potential algorithm with annealed kernel sizes and the convolution smoothing method in global optimization. This property is extremely significant since it demonstrates that there is a built-in parameter in the algorithm itself (the kernel size) to achieve global optimization, even when gradient descent is utilized. Some simulation results will be presented in the following to support this conjecture.

VI. TDNN Training Example

We now consider a numerical case study, specifically, TDNN training with information potential maximization criterion for single-step prediction of the Mackey–Glass (MG) time series [30]. This example is a continuation of the results presented in [18], [19] with quadratic entropy and Gaussian kernels and was chosen to show the practical impact of the enhancements developed in this paper. The MG time series is generated by an MG system with delay parameter $\tau = 30$. The MLP input vector consists of six consecutive samples of the MG time series to be consistent with Taken's embedding theorem [31] and the desired output is designated as the following sample from the sequence. A training set of a mere 200 input–output pairs is prepared in this manner (roughly two cycles of the trajectory over the attractor). The TDNN consists of six processing elements (PEs) in a hidden layer with biases and $\tanh$ nonlinearities and a single linear output PE. Since the information potential does not change with the mean of the error pdf, the bias value of the linear output PE was set to match the mean of the desired output after the training of the other weights had concluded so that the error mean is zero over the training set. Gaussian kernels are used throughout the examples.

Example I: Comparison With MSE: First, we consider the case where the Gaussian kernel size is kept fixed at $\sigma = 10^{-3}$ during the adaptation and the entropy order is $\alpha = 2$. In this case, to avoid local-optima, each TDNN is trained starting from 1000 predetermined initial sets of weights, which were generated by a uniform distribution in the interval $[-1,1]$. Then, the best solution, i.e., the one with the highest information potential after convergence, among the 1000 candidates was selected for each TDNN. A similar procedure was followed using the MSE criterion for the same initial conditions. Fig. 2 shows that the TDNN trained using the entropy criterion achieved a better pdf fit to the distribution of the desired signal data compared to the TDNN trained using MSE. The pdfs of the original and predicted time series are evaluated using Parzen windowing over a test set of 10 000 samples. We can observe that the entropy training is a better fit to the pdf of the original MG system. See [18] and [19] for further details.

Example 2: Effect of α and Kernel Size: In a second set of simulations, the solutions obtained by using different Gaussian kernel sizes and entropy orders are compared. For each set of parameters, 100 initial conditions were utilized. After the training,

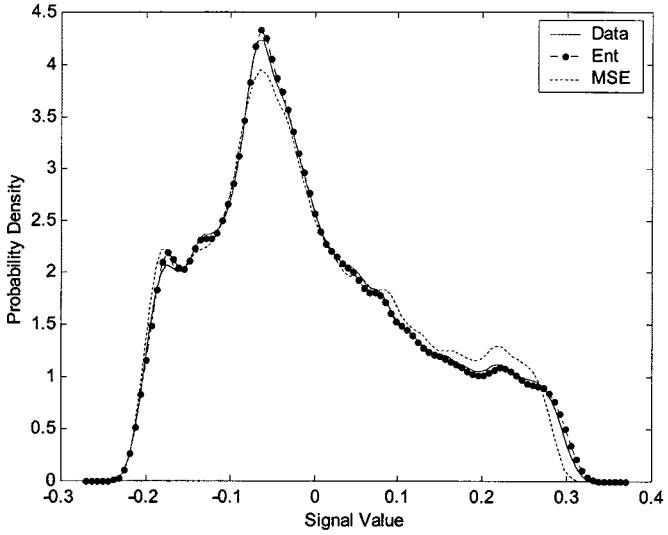

Fig. 2. Comparison of pdf fits achieved by entropy and MSE criteria; desired signal (solid), entropy solution (dots), MSE solution (dotted).

TABLE I
EVALUATING THE NORMALIZED INFORMATION POTENTIAL OF ERROR SAMPLES AT DIFFERENT α VALUES (MAXIMUM POSSIBLE VALUE IS 1) FOR TDNNS TRAINED WITH DIFFERENT PARAMETERS

Training parameters \ Evaluation parameters		$V_\alpha(e)$ α=1.01 $\sigma=10^{-3}$	$V_\alpha(e)$ α=1.5 $\sigma=10^{-3}$	$V_\alpha(e)$ α=2 $\sigma=10^{-3}$	$V_\alpha(e)$ α=3 $\sigma=10^{-3}$
α=1.01	σ=0.01	*0.976*	0.304	0.099	0.012
	σ=0.1	*0.976*	0.311	0.104	0.013
	σ=1	*0.969*	0.212	0.047	0.002
α=1.5	σ=0.01	0.977	*0.321*	0.112	0.016
	σ=0.1	0.977	*0.318*	0.109	0.015
	σ=1	0.976	*0.312*	0.105	0.014
α=2	σ=0.01	**0.979**	**0.352**	**0.135**	**0.023**
	σ=0.1	**0.979**	**0.352**	**0.133**	**0.021**
	σ=1	**0.978**	**0.343**	**0.126**	**0.019**
α=3	σ=0.01	0.977	0.336	0.124	*0.020*
	σ=0.1	0.977	0.330	0.117	*0.017*
	σ=1	0.976	0.312	0.105	*0.014*

the information potential of the error on the test set (consisting of 10 000 samples) corresponding to each TDNNs for different α and σ are evaluated using Gaussian kernels with $\sigma = 10^{-3}$ (for the final error signals, each kernel covers an average of about ten samples for this kernel size). This value is chosen because it is at least one order of magnitude smaller than any of the kernel size values utilized in training and, therefore, allows a fair comparison of the results obtained through different cost function parameters. Table I shows a comparison of the performances of TDNNs trained with different parameters of the criterion. There are a total of 12 trained TDNNs, using the designated entropy orders and kernel sizes given in the first column. The performances of these TDNNs are then evaluated and compared using four different entropy orders, presented in each column.

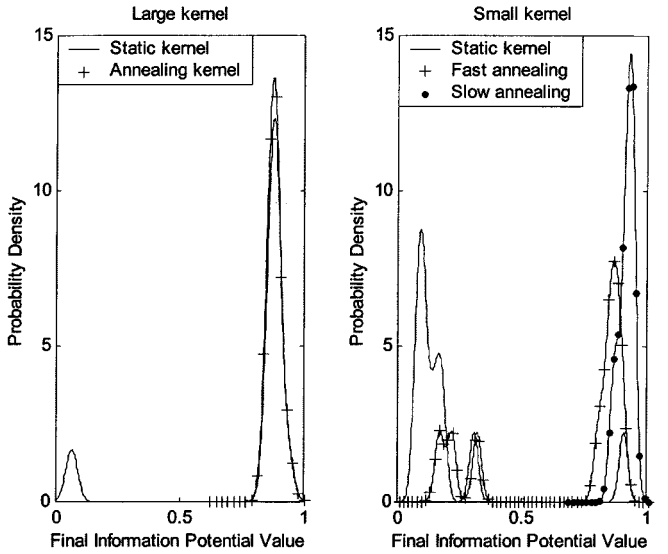

Fig. 3. Probability distribution of final normalized information potential when kernel size is (a) large: static kernel (solid), slow annealing $(+)$; (b) small: static kernel (solid), fast annealing $(+)$, slow annealing (dots).

When inspecting these results, the data in each column should be compared. Each row corresponds to the performance of the TDNN trained using the parameter values designated with each column giving the evaluation of the information potential for this data using different entropy orders.

Notice that, regardless of the entropy order used in evaluation (each column), the TDNN trained using the quadratic entropy $(\alpha = 2)$ yields the best performance. Furthermore, using smaller kernel sizes in training also improves performance slightly.

Example 3: Annealing the Kernel Size: The third example presents results from a set of simulations where the kernel size is annealed during training from a large value to a smaller one. The results of these experiments are summarized in Fig. 3. In Fig. 3(a), the probability distributions of the final information potential values (normalized such that the maximum possible value is one, when all the samples are equal to each other) obtained with 100 random initial conditions for two experiments (fixed and annealed kernels) are shown. The same training window size of 200 samples as in Example 1 is used here and the information potential is estimated with Gaussian kernels and quadratic entropy. In the static kernel case, the kernel size is kept fixed at $\sigma = 10^{-2}$, whereas the annealed kernel had an exponentially decreasing kernel size $\sigma = 10^{-1} \rightarrow 10^{-2}$, during a training phase of 200 iterations with quadratic entropy and 200 samples. It is clear that, for this large kernel size of $\sigma = 10^{-2}$, the static kernel sometimes (10% of the time) gets trapped in a local maximum of the information potential (which has a normalized value of about 0.1). The annealed kernel avoids the local optimum in all the runs and achieves the global maximum (which has a normalized value of about 0.9). In Fig. 3(b), the distributions of the performances for three experiments are shown, but now the static kernel has a size of $\sigma = 10^{-3}$ throughout the training. We can expect more local maxima with this smaller kernel value, but more accurate performance if global maximum is achieved (due to results in Table I). The slow and fast annealed kernels, on the other hand,

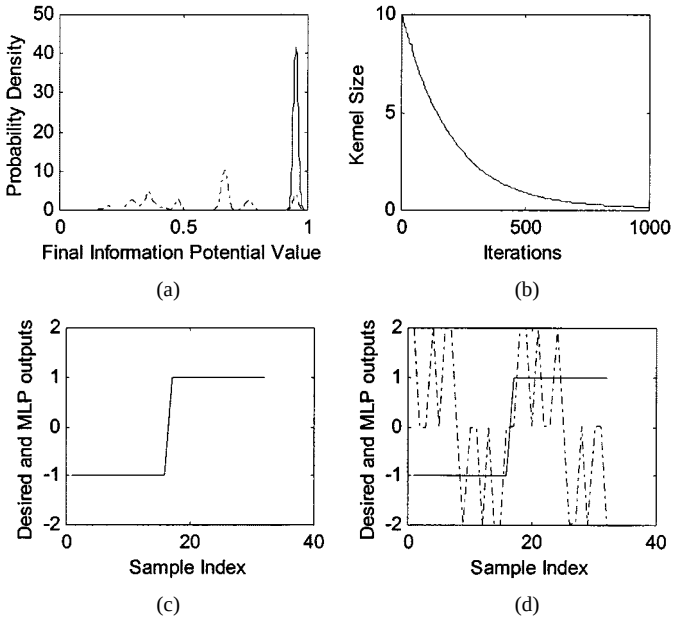

Fig. 4. Results for XOR problem. (a) Distributions of final (normalized) information potential values, static kernel (dotted), annealed kernel (solid). (b) Annealing of the kernel size vs iterations. (c) A sample from annealed kernel case, desired output (solid), MLP output (dotted). (d) A local optimum sample from static kernel case, desired output (solid), MLP output (dotted).

have exponentially decreasing sizes of $\sigma = 10^{-1} \rightarrow 10^{-3}$ for a training phase of 500 and 200 iterations, respectively. This annealing scheme is the same for all initial conditions. In this small kernel case with $\sigma = 10^{-3}$, it is observed that the static kernel gets trapped in local maxima quite often (90% of the time), whereas the fast annealed kernel demonstrates some improvement in terms of avoiding local optima (70% of the time achieves global optimum) and eventually the slow annealed kernel consistently achieves the global maximum (100% of the time). These experiments have demonstrated that, by annealing the kernel size, one is likely to improve the algorithm's chances of avoiding local optima. However, there is no prescription for how to anneal the kernels, yet. The exponential annealing scheme and the decay rates were determined by trial and error.

The second numerical case study we present is a classification problem. Namely, the five-bit parity problem in the class of generalized XOR problems is considered. In this case study, a 5-5-1 MLP is utilized with $\tanh$ nonlinearities in the hidden layer and a linear output PE. The five inputs take the values ± 1 according to the considered bit sequence and the desired output is again ± 1, the XOR value corresponding to the input sequence. The training set consists of all possible input sequences, numbering 32. In the static kernel case, the kernel size is set to $\sigma = 10^{-1}$ and the MLP is trained for 1000 iterations starting from 100 random initial weight vectors. In the annealed kernel case, the kernel size is annealed down exponentially as $\sigma = 10 \rightarrow 10^{-1}$ in 1000 iterations. The MLP is trained starting from the same 100 initial weight vectors. It has been observed that the MLP trained using annealed kernels achieved global optimum in all trials (100% of the time), whereas the MLP trained using the static kernels could rarely achieve the global optimum (10% of the time). The results of these experiments are summarized in Fig. 4. In Fig. 4(a), the probability distribution of the final (normalized) information potential values is presented. It is clear that with the annealed kernels, the final information potential values are concentrated around the global maxium, whereas, with the static kernels, the algorithm is trapped at local maxima often. Fig. 4(b), shows how the kernel size is exponentially annealed down in 1000 iterations. Figs. 4(c) and (d) are samples of outputs of MLPs with the set of optimal weights that match the output to the desired exactly and with a set of local optimum weights that produce an extremely low grade output.

VII. Conclusion

Renyi's entropy of the error signal in supervised learning was previously proposed as an alternative to MSE and it was shown in a number of applications that the quadratic entropy had advantages over this conventional performance index. Initially, the main focus was on the special case of quadratic entropy with Gaussian kernels due to the analytical simplifications gained with these choices. In this paper, we have proposed an enhanced, more flexible approach to estimate nonparametrically the Renyi's entropy of a random variable from its samples. We have shown that this new estimator is equivalent to the previously suggested entropy estimator, if the entropy order is set to two and the kernel function is the Gaussian function.

Since our estimator employs Parzen windowing to estimate the pdf of a random variable from the samples, we investigated the question whether this process preserved the global minima of the actual quantity and showed that, in fact, this is the case. Furthermore, we have extended to all orders of entropy, the information potential and information force concepts, which were previously defined for only the quadratic entropy case. We have established the links between the order-α information force and potential and their quadratic special cases. We have also addressed the possible advantages of configuring the entropy order according to the peakiness of the data pdf whenever this information is available. If not, the choice of quadratic entropy seems the most appropriate and we can consider $\alpha = 2$ as the natural choice for entropy estimation with the information potential. Interestingly enough, in terms of computational savings, quadratic entropy is also the most advantageous. The choice of alternative kernels for entropy estimation was not addressed in this paper, but represents still another avenue for further research.

Another very important aspect of the proposed Renyi's entropy criterion is its close relation with the convolution smoothing method of global optimization. We have explored the effect of the kernel size on the criterion and the search algorithm and came up with the very important understanding that it is also possible to use this design parameter to our advantage in the training process, i.e., by starting with a large kernel size and properly decreasing it to avoid local-optimum solutions with gradient rules. The final value of the kernel size should not be zero but a predetermined nominal value, which describes the right balance between the estimation bias and the variance for the specific problem at hand. We are currently investigating the choice of appropriate minimal kernel for entropy estimation.

Finally, we have applied the criterion to the problem of TDNN training in a short-term chaotic time series prediction

problem. In this problem, we have investigated the performance of the solutions generated by TDNNs that are trained using different orders of entropy and different kernel sizes. Simulation results suggested that in fact the quadratic entropy might produce the best solutions. More analytical and numerical studies are needed to determine which order of entropy is most suitable for which problem. Similar analyses for the generalized XOR problem had been carried out. It has been confirmed that annealing the kernel size down from a large value helps achieve global optima.

APPENDIX A

Proof of Lemma 1: Let $\bar{e} = [e_1 \ \cdots \ e_N]^T$ be the error vector over the training data set. To prove this statement, we evaluate the gradient and the Hessian of the entropy estimator at the point $\bar{e} = 0$, without loss of generality. The gradient and the Hessian of the entropy estimator with respect to $\bar{e}$ can be written in terms of the gradient and Hessian of the information potential as follows:

$$\frac{\partial \hat{H}_\alpha}{\partial e_k} = \frac{1}{1-\alpha}\frac{\partial \hat{V}_\alpha/\partial e_k}{\hat{V}_\alpha}$$
$$\frac{\partial^2 \hat{H}_\alpha}{\partial e_l \partial e_k} = \frac{1}{1-\alpha}\frac{(\partial^2 \hat{V}_\alpha/\partial e_l \partial e_k)\hat{V}_\alpha - (\partial \hat{V}_\alpha/\partial e_k)(\partial \hat{V}_\alpha/\partial e_l)}{\hat{V}_\alpha^2}. \tag{A.1}$$

The following can be verified with trivial derivation and substitution of $\bar{e} = 0$

$$\begin{aligned}
\hat{V}_\alpha\big|_{\bar{e}=0} &= \kappa_\sigma^{\alpha-1}(0) \\
\left.\frac{\partial \hat{V}_\alpha}{\partial e_k}\right|_{\bar{e}=0} &= \frac{(\alpha-1)}{N^\alpha}\left[N^{\alpha-1}\kappa_\sigma^{\alpha-2}(0)\kappa'(0)\right. \\
&\quad \left. - N^{\alpha-1}\kappa_\sigma^{\alpha-2}(0)\kappa'(0)\right] = 0 \\
\left.\frac{\partial^2 \hat{V}_\alpha}{\partial e_k^2}\right|_{\bar{e}=0} &= \frac{(\alpha-1)(N-1)\kappa_\sigma^{\alpha-3}(0)}{N^2} \\
&\quad \times [(\alpha-2)\kappa'^2(0) + 2\kappa(0)\kappa''(0)] \\
\left.\frac{\partial^2 \hat{V}_\alpha}{\partial e_l \partial e_k}\right|_{\bar{e}=0} &= -\frac{(\alpha-1)\kappa_\sigma^{\alpha-3}(0)}{N^2} \\
&\quad \times [(\alpha-2)\kappa'^2(0) + 2\kappa(0)\kappa''(0)].
\end{aligned} \tag{A.2}$$

Hence the entries of the Hessian matrix are shown in (A.3) at the bottom of the page, and the eigenvalue-eigenvector pairs are

$$\{0, [1, \ldots, 1]^T\}, \{aN/(N-1), [1, -1, 0, \ldots, 0]^T\} \\ \{aN/(N-1), [1, 0, -1, 0, \ldots, 0]^T\}, \ldots. \tag{A.4}$$

The zero eigenvalue and the corresponding eigenvector are due to the fact that the entropy is invariant with respect to changes in the mean of the random variable. Thus, the entropy has a minimum line instead of a single point along the direction where only the mean of the error samples changes. Provided that $\kappa'(0) = 0$ (which is the case for symmetric and differentiable kernels), the nonzero eigenvalue with multiplicity $(N-1)$ at $\bar{e} = 0$ is positive iff $N > 1, \kappa(0) > 0$, and $\kappa''(0) < 0$. A symmetric, continuous and differentiable, and unimodal kernel satisfies these requirements. □

Proof of Theorem 1: Assume we use a kernel function as described in the theorem. In order to prove this statement, consider the value of the entropy estimator in (3) when all the samples are identical. In that case, the kernel evaluations are all performed at zero and the entropy becomes $H_\alpha(0) = -\log \kappa_\sigma(0)$, independent of entropy order. We need to show that

$$\frac{1}{1-\alpha}\log\left[\frac{1}{N^\alpha}\sum_j\left(\sum_i \kappa_\sigma(e_j - e_i)\right)^{\alpha-1}\right] \geq -\log \kappa_\sigma(0). \tag{A.5}$$

For $\alpha > 1$, this is equivalent to showing that

$$\sum_j\left(\sum_i \kappa_\sigma(e_j - e_i)\right)^{\alpha-1} \leq N^\alpha \kappa_\sigma^{\alpha-1}(0). \tag{A.6}$$

We start by the expression on the left-hand side and replace terms by their upper bounds

$$\begin{aligned}
&\sum_j\left(\sum_i \kappa_\sigma(e_j - e_i)\right)^{\alpha-1} \\
&\quad \leq N \max_j\left[\left(\sum_i \kappa_\sigma(e_j - e_i)\right)^{\alpha-1}\right] \\
&\quad \leq N \max_j\left[N^{\alpha-1}\max_i \kappa_\sigma^{\alpha-1}(e_j - e_i)\right] \\
&\quad = N^\alpha \max_{i,j} \kappa_\sigma^{\alpha-1}(e_j - e_i) \leq N^\alpha \kappa_\sigma^{\alpha-1}(0).
\end{aligned} \tag{A.7}$$

This completes the proof for the case $\alpha > 1$. The proof for $\alpha < 1$ is similar. It uses the min operator instead of maxdue to direction of inequality. □

Proof of Theorem 2: The error is given as the difference between the desired output and the actual output, i.e., $e = d - y$. Using this identity, we can relate the pdf of error to the pdf of the output as $f_{e,w}(e) = f_{y|x,w}(d - e|x)$, where the subscript w denotes dependence on the optimization parameters.

$$\left.\frac{\partial^2 \hat{H}_\alpha}{\partial e_l \partial e_k}\right|_{\bar{e}=0} = \begin{cases} a \triangleq -(N-1)\kappa_\sigma^{-\alpha-1}(0)[(\alpha-2)\kappa'^2(0) + 2\kappa(0)\kappa''(0)]/N^2, & l = k \\ b \triangleq \kappa_\sigma^{-\alpha-1}(0)[(\alpha-2)\kappa'^2(0) + 2\kappa(0)\kappa''(0)]/N^2, & l \neq k \end{cases} \tag{A.3}$$

Minimum error entropy problem is formulated as follows (the integral limits are from $-\infty$ to ∞):

$$\begin{aligned}\min_w \frac{1}{1-\alpha}\log\int f_{e,w}^{\alpha}(e)\,de &= \frac{1}{1-\alpha}\log\int f_{y\,|\,x,w}^{\alpha}(d-e\,|\,x)\,de \\ &= \frac{1}{1-\alpha}\log\int -f_{y\,|\,x,w}^{\alpha}(y\,|\,x)\,dy \\ &\qquad \text{(variable change } y=d-e\text{)}. \quad \text{(A.8)}\end{aligned}$$

In concern for space, we continue here for the case $\alpha > 1$ only. In that case

$$\begin{aligned}&\equiv \min_w \int f_{y\,|\,x,w}^{\alpha}(y\,|\,x)\,dy \cdot \int f_x^{\alpha}(x)\,dx \\ &= \int\int f_{xy,w}^{\alpha}(x,y)\,dx\,dy \\ &\equiv \int\int f_{xy,w}^{\alpha}(x,y)\,dx\,dy \cdot \int\int f_{xd}^{1-\alpha}(x,y)\,dx\,dy \\ &= \min_w \int\int f_{xy,w}(x,y)\left(\frac{f_{xd}(x,y)}{f_{xy,w}(x,y)}\right)^{1-\alpha} dx\,dy. \quad \text{(A9)}\end{aligned}$$

We recognize this final expression as the Csiszar distance between the joint pdfs of input-desired and input–output signals. In particular, the convex function in Csiszar distance is chosen here to be $(\cdot)^{1-\alpha}$, which is convex for $\alpha > 1$. In order to see this distance measure reduce to the Kullback–Leibler (KL) divergence, consider the following modification. Minimizing the Csiszar distance above is equivalent to minimizing

$$\min_w \frac{1}{\alpha-1}\log\int\int f_{xy,w}(x,y)\left(\frac{f_{xd}(x,y)}{f_{xy,w}(x,y)}\right)^{1-\alpha} dx\,dy \quad \text{(A.10)}$$

since $\alpha > 1$ and log is monotonic. Taking the limit of this expression as $\alpha \to 1^{+}$ using L'Hopital's rule yields the KL distance measure. In fact, starting the derivation from Shannon's entropy definition for the error, one arrives directly at the KL divergence. □

APPENDIX B

In this Appendix, we prove that an $h_\beta(w)$ exists, which guarantees the equivalence desired and also show that it satisfies some of the required conditions. The desired $h_\beta(w)$ satisfies the following equality:

$$h_\beta(w)^*_w \int f_e^{\alpha}(e;w)\,de = \int [f_e(e;w)^*_e \kappa_\sigma(e)]^{\alpha}\,de. \quad \text{(B.1)}$$

Taking the Laplace transform of both sides with respect to w, we can isolate the Laplace transform of $h_\beta(w)$ in terms of the transforms of the remaining quantities. The Laplace transform of $h_\beta(w)$ is guaranteed to exist if the error pdf and the kernel function are absolutely integrable functions and $\alpha \geq 1$, which is the case. We can write this transformed function as the following ratio. The right-hand side is a function of s only since the integration over e from $-\infty$ to ∞ eliminates this variable

$$\begin{aligned}H_\beta(s) &= \frac{L_w \int [f_e(e;w)^*_e \kappa_\sigma(e)]^{\alpha}\,de}{L_w \int f_e^{\alpha}(e;w)\,de} \\ &= \frac{\int L_w [f_e(e;w)^*_e \kappa_\sigma(e)]^{\alpha}\,de}{\int L_w\,[f_e^{\alpha}(e;w)]\,de}. \quad \text{(B.2)}\end{aligned}$$

Since $H_\beta(s)$ exists, $h_\beta(w)$ must be absolutely integrable, therefore, $\lim_{w\to\pm\infty} h_\beta(w) = 0$. We next observe that as $\sigma \to 0$, the numerator of (B.2) converges to the denominator, hence, the bandwidth of $H_\beta(\omega)$ (considering the Fourier transform) increases. An increase in frequency-domain is accompanied by a decrease in duration of the impulse response in time-domain, thus, the width of $h_\beta(w)$ decreases as $\sigma \to 0$ and there is a nonlinear monotonous relation between β and σ. Now that we know the width of $h_\beta(w)$ decreases monotonously as $\sigma \to 0$, that it is always absolutely integrable, and that it converges to $\delta(w)$ in the limit, we conclude that it has to be unimodal, symmetric, and positive for all w. Consequently, even if $h_\beta(w)$ does not integrate to one, it integrates to some finite value and, therefore, it is a scaled version of a pdf. A scaling factor in the convolution process does not affect the nature of the smoothing but only the scale factor that multiplies the smoothed performance surface.

REFERENCES

[1] G. Deco and D. Obradovic, *An Information-Theoretic Approach to Neural Computing*. New York: Springer-Verlag, 1996.

[2] R. Linsker, "Toward an organizing principle for a layered perceptual network," in *Neural Information Processing Systems*, D. Anderson, Ed. New York: Amer. Inst. Phys., 1988, pp. 485–494.

[3] J. Kapur and H. Kesavan, *Entropy Optimization Principles and Applications*: Associated Press, 1992.

[4] R. V. Hartley, "Transmission of information," *Bell Syst. Tech. J.*, vol. 7, 1928.

[5] C. E. Shannon, "A mathematical theory of communications," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, 623–656, 1948.

[6] S. Amari, *Differential-Geometrical Methods in Statistics*. Berlin, Germany: Springer-Verlag, 1985.

[7] P. Viola, N. Schraudolph, and T. Sejnowski, "Empirical entropy manipulation for real-world problems," in *Proc. Neural Inform. Processing Syst. (NIPS 8) Conf.*, 1995, pp. 851–857.

[8] P. Comon, "Independent component analysis, a new concept?," *Signal Processing*, vol. 36, pp. 287–314, 1994.

[9] H. Yang and S. Amari, "Adaptive online learning algorithms for blind separation: Maximum entropy and minimum mutual information," *Neural Comput.*, vol. 9, pp. 1457–1482, 1997.

[10] E. Parzen, "On estimation of a probability density function and mode," in *Time Series Analysis Papers*. San Francisco, CA: Holden-Day, 1967.

[11] J. C. Principe, D. Xu, and J. Fisher, "Information theoretic learning," in *Unsupervised Adaptive Filtering*, S. Haykin, Ed. New York: Wiley, 2000, vol. I, pp. 265–319.

[12] K. E. Hild II, D. Erdogmus, and J. C. Principe, "Blind source separation using Renyi's mutual information," *IEEE Signal Processing Lett.*, vol. 8, pp. 174–176, June 2001.

[13] D. Xu, J. C. Principe, J. Fisher, and H. Wu, "A novel measure for independent component analysis (ICA)," in *Proc. ICASSP'98*, vol. II, Seattle, WA, pp. 1161–1164.

[14] K. Torkkola and W. M. Campbell, "Mutual information in learning feature transformations," presented at the Proc. Int. Conf. Machine Learning, Stanford, CA, 2000.

[15] V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.

[16] C. Diks, J. Houwelingen, F. Takens, and J. deGoede, "Detecting differences between delay vector distributions," *Phys. Rev. E*, vol. 53, pp. 2169–2176, 1996.

[17] P. Grassberger and I. Procaccia, "Characterization of strange attractors," *Phys. Rev. Lett.*, vol. 50, no. 5, pp. 346–349, 1983.
[18] D. Erdogmus and J. C. Principe, "Comparison of entropy and mean square error criteria in adaptive system training using higher order statistics," presented at the Proc. Independent Components Analysis (ICA), Helsinki, Finland, 2000.
[19] ——, "An entropy minimization algorithm for short-term prediction of chaotic time series," IEEE Trans. Signal Processing, vol. 50, pp. 1780–1786, July 2002, submitted for publication.
[20] A. Renyi, *Probability Theory*. New York: Elsevier, 1970.
[21] I. Csiszár and J. Körner, *Information Theory: Coding Theorems for Discrete Memoryless Systems*. New York: Academic, 1981.
[22] D. Erdogmus, K. E. Hild II, and J. C. Principe, "Blind source separation using Renyi's marginal entropy," *Neurocomputing (Special Issue on Blind Source Separation and Independent Component Analysis)*, 2001.
[23] S. C. Douglas and S.-Y. Kung, "Kuicnet algorithms for blind deconvolution," in *Proc. Neural Networks Signal Processing VIII (NNSP'98)*, Cambridge, U.K., 1998, pp. 3–12.
[24] S. Haykin, *Introduction to Adaptive Filters*. New York: MacMillan, 1984.
[25] D. Rumelhart, G. Hinton, and R. Williams, "Learning internal representations by error back-propagation," *Nature*, vol. 323, pp. 533–536, 1986.
[26] D. G. Luenberger, *Linear and Nonlinear Programming*. Reading, MA: Addison-Wesley, 1973.
[27] A. Aarts and J. Korst, *Simulated Annealing and Boltzmann Machines: A Stochastic Approach to Combinatorial Optimization and Neural Computing*. New York: Wiley, 1989.
[28] R. Y. Rubinstein, *Simulation and the Monte Carlo Method*. New York: Wiley, 1981.
[29] W. Edmonson, K. Srinivasan, C. Wang, and J. Principe, "A global least square algorithm for adaptive IIR filtering," *IEEE Trans. Circuits Syst.*, vol. 45, pp. 379–384, Mar. 1996.
[30] D. Kaplan and L. Glass, *Understanding Nonlinear Dynamics*. New York: Springer-Verlag, 1995.
[31] J. M. Kuo, "Nonlinear Dynamic Modeling with Artificial Neural Networks," Ph.D. dissertation, Univ. Florida, Gainesville, 1993.

Deniz Erdogmus (S'95) received the B.S. degrees in electrical and electronics engineering and mathematics in 1997 and the M.S. degree in electrical and electronics engineering, with emphasis on systems and control, in 1999, all from the Middle East Technical University, Ankara, Turkey. Since 1999, he has been pursuing the Ph.D. degree at the Electrical and Computer Engineering Department at University of Florida, Gainesville.

From 1997 to 1999, he worked as a Research Engineer with the Defense Industries Research and Development Institute (SAGE) under the Scientific and Technical Research Council of Turkey (TUBITAK). His current research interests include information theory and its applications to adaptive systems and adaptive systems for signal processing, communications and control.

Mr. Erdogmus is a Member of Tau Beta Pi and Eta Kappa Nu.

Jose C. Principe (F'00) received the Licenciatura degree in electrical engineering from the University of Porto, Portugal, in 1972 and the M.Sc. and Ph.D. degrees in electrical engineering from the University of Florida, Gainesville, in 1979 and 1985, respectively.

He is Professor of Electrical and Computer Engineering and Biomedical Engineering at the University of Florida, where he teaches advanced signal processing, machine learning, and artificial neural networks (ANNs) modeling. He is the BellSouth Professor and the Founder and Director of the University of Florida Computational NeuroEngineering Laboratory (CNEL). He has more than 70 publications in refereed journals, ten book chapters, and 160 conference papers. He directed 35 Ph.D. dissertations and 45 Master's theses. He recently wrote an interactive electronic book entitled, *Neural and Adaptive Systems: Fundamentals Through Simulation* (New York: Wiley, 2000). His primary area of interest is the processing of time-varying signals with adaptive neural models. The CNEL Lab has been studying signal and pattern recognition principles based on information theoretic criteria (entropy and mutual information).

Dr. Principe is the Chair of the Technical Committee on Neural Networks of the IEEE Signal Processing Society, a Member of the Board of Governors of the International Neural Network Society, and Editor-in-Chief of the IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING. He is a member of the Advisory Board of the University of Florida Brain Institute.