Signal Classification through Multifractal Analysis and Complex Domain Neural Networks

W. Kinsner, V. Cheung, K. Cannons, J. Pear*, and T. Martin*
Department of Electrical and Computer Engineering
Signal and Data Compression Laboratory
**Department of Psychology*
University of Manitoba, Winnipeg, Manitoba, Canada, R3T 5V6
email: kinsner@ee.umanitoba.ca, {umcheu10 | umcanno0 | pear | ummarti7}@cc.umanitoba.ca

Abstract

This paper describes a system capable of classifying stochastic, self-affine, nonstationary signals produced by nonlinear systems. The classification and analysis of these signals is important because they are generated by many real-world processes. The first stage of the signal classification process entails the transformation of the signal into the multifractal dimension domain, through the computation of the variance fractal dimension trajectory (VFDT). Features can then be extracted from the VFDT using a Kohonen self-organizing feature map. The second stage involves the use of a complex domain neural network and a probabilistic neural network to determine the class of a signal based on these extracted features. The results of this paper show that these techniques can be successful in creating a classification system which can obtain correct classification rates of about 87% when performing classification of such signals without knowing the number of classes.

Keywords: *Multifractal analysis; complex domain neural networks; probabilistic neural networks; classification*

1. Introduction

This paper investigates the development of a software system that is capable of classifying stochastic, self-affine, nonstationary signals that originate from nonlinear systems. Such signals are often multivariate, and the system described in this paper has the ability to take these multivariate signals into account during the classification process.

The features used for classification are based on a temporal multifractal characterization of the signal, which is achieved through the computation of its variance fractal dimension trajectory (VFDT) [Kins94]. This translation into the temporal multifractal dimension domain emphasizes the underlying complexity of the signal, and more importantly for classification, has a normalizing effect. The classification based on these features is performed by a complex domain neural network that can operate upon signal features from separate, but strongly correlated signals, without losing the correlation between the signals. Furthermore, complex domain neural networks often generalize more effectively and train faster than their real-valued counterparts.

While the classification system implemented for this paper is not specific to any particular signal, spatio-temporal recordings of a Siamese fighting fish when presented with various stimuli during dishabituation experiments were used to evaluate the performance of the system. A stereoscopic camera system was used to track and record the three dimensional Cartesian co-ordinates of the fish over an eight hour period. A sampling rate of 10 Hz was used for these recordings, which is valid according to the Nyquist sampling theorem and the known physical limits of the fish [ChCa03]. Each sample in the signal was recorded with an accuracy of two decimal places. Finally, the recording device performed no filtering of the signal and thus, did not alter its bandwidth.

A sample of this dishabituation signal is shown in Fig. 1. Stimuli applied during these experiments were on the Y-Z plane at X = 0 mm and on the X-Y plane at approximately Z = 225 mm. Since there were no stimuli along the Y-axis and it was the least accurate because it was resolved indirectly through the stereoscopic vision, the Y-component of the signal was not used for classification in this paper [ChCa03]. An added difficulty in analyzing these signals was that they contained an

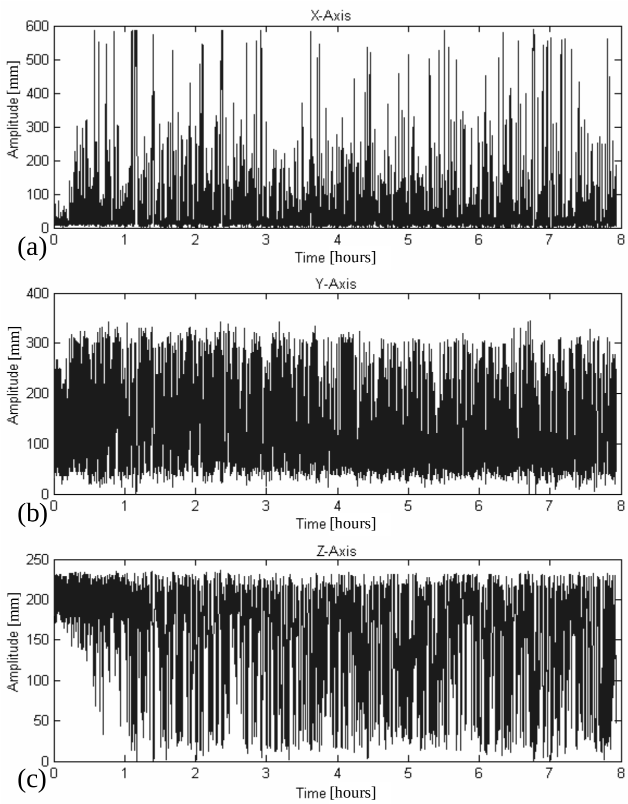

Fig. 1. Fish trajectory signal along the X (a), Y (b), and Z(c) directions.

unknown number of classes, but this was overcome using clustering algorithms.

An overview of the desired behavioural analysis and the techniques used in this paper for classification of the behavioural classes is provided in Sec. 2. Details of the experiments performed with the dishabituation signals and the classification system are presented in Sec. 3.

2. Background

2.1 Animal Behavioural Analysis

Experimental analysis is an effective means of discovering laws that govern the interaction between animal behaviour and its environment. Because behaviour is a continuous phenomenon, this effort requires that we subdivide or classify behaviour in some way.

One way to form response classes is to create contingencies that are satisfied by a variety of response topographies, and to study selected properties of any behaviour that satisfies those contingencies. For example, a rat in an experimental chamber may be trained to depress a lever, and the frequency of this behaviour will be a function of which lever presses, if any, produce access to food. This approach of defining *functional* response classes has helped to produce powerful and general learning principles, but the effects of the experimental contingencies on other properties of the behaviour often go unstudied. That is, the rat can do many different things that will lower the lever, and may be engaged in a variety of activities between lever presses.

The computer-aided video tracking system described here measures behaviour in a way that captures its continuity and diverse topographies, but the problem of classification remains: how do we efficiently extract interesting and relevant portions of behaviour and relate these behaviours to environmental events? The system can easily generate so much data that it becomes very difficult to try to identify recurring patterns through visual inspection. The techniques described in this paper are an important step towards the automated identification and analysis of *topographical* response classes. Combining this data with analysis of functional response classes is certain to help produce a more complete science of animal behaviour.

2.2 Variance Fractal Dimension Trajectory

The feature extraction technique used in this paper for classification is a transformation into the temporal multifractal dimension domain by computing a variance fractal dimension trajectory [Kins94]. An advantage of using the variance fractal dimension trajectory (VFDT) for classification is that it emphasizes the underlying complexity of the signal, thus helping to provide the unique identification for each class. Another advantage is that the transformation provides a normalizing effect because the theoretical range of fractional dimensionality of Euclidean one-dimensional signal is between 1 and 2. A final important advantage of the VFDT is that it can be computed in real-time, which significantly broadens its possible applications.

Before proceeding with a more detailed description of the VFDT, the concepts of fractals and fractal dimensions will first be explained. The fish trajectory signals which are being examined in this paper are, among other things, self-affine in nature. Self-affinity means that regardless of the magnification used when viewing an object, the object's statistical properties, structure, and complexity remain constant [Mand82]. Fractals, in the simplest sense, are self-affine entities. As a result, fractal analysis techniques serve as an appropriate means for processing the signals in this paper.

An important characteristic of a fractal is its fractal dimension. The most familiar dimensions are the standard Euclidean dimensions, which are discrete, integral numbers. For instance, a line is considered to be a one dimensional object, a square a two dimensional object, and a cube a three dimensional object. However, it is also possible for objects to exhibit fractional dimensions. In fact, the term fractals was coined to describe objects that have a dimension that is non-integral. The Koch curve, for example, is a fractal curve with a dimension of approximately 1.2619 [PeJS92], meaning that it has a greater complexity than a straight line, but is not quite as complex as a two dimensional object.

The significance of the dimensionality of an object is that it provides valuable information regarding the object's complexity, which in this context, refers to the singularities in the object. There are an infinite number of fractal dimensions, such as the topological dimension, box-counting dimension, self-similar dimension, and information dimension. In certain instances, the different measures of dimensionality yield the same result for the same object. However, at other times, the computation of the various dimensions provides different numerical values, which leads to the notion of multifractals. For the purposes of this paper, only one form of fractal dimension, the variance fractal dimension [Kins94], is utilized to perform feature extraction.

The variance fractal dimension is based on calculations involving the variance of the amplitude increments of a signal taken at different scales. The amplitude increments of a signal over a time interval Δt adhere to the following power law relationship

$$Var[x(t_2) - x(t_1)] \sim |t_2 - t_1|^{2H} \qquad (1)$$

where $x(t)$ represents the signal and H is the Hurst exponent. The Hurst exponent can be calculated via a log-log plot using

$$H = \lim_{\Delta t \to 0} \frac{1}{2} \frac{\log[Var(\Delta x)_{\Delta t}]}{\log(\Delta t)} \qquad (2)$$

The variance fractal dimension, D_σ, is then given by

$$D_\sigma = E + 1 - H \qquad (3)$$

where E is the Euclidean dimension. The Euclidean dimension is equal to the number of independent variables in the signal. Thus, since this paper concentrates solely upon Euclidean one-dimensional signals, E can be set to 1. Equation 3 then reduces to:

$$D_\sigma = 2 - H \qquad (4)$$

The process of calculating the VFDT of a signal essentially involves segmenting the entire signal into numerous sub-signals, or windows, and calculating the variance fractal dimension for each of these windows. The choice of the window size is a very important aspect of the process and is specific to the type of signal under analysis. Generally, the window size is chosen to match the stationarity of the signal being studied. A second important parameter of the VFDT is the window displacement, which is the number of samples that the window is shifted for each calculation of the variance fractal dimension. If the window displacement is selected such that it is smaller than the window size, then the windows used in calculating the VFDT will overlap. It is not necessary that the windows overlap; however, using overlapping windows amplifies the details of the VFDT because the same points in the original signal are involved in multiple windows while computing the VFDT. However, the displacement should not be given an extremely small value since this would result in significant windowing artifacts because of the excessive correlation between windows. Of the two parameters, the determination of the window displacement is more subjective in nature and usually determined experimentally.

This sliding window computation is significant in that it reveals the changes in the variance fractal dimension of the signal over time, which is shown in this paper to provide valuable information for classification purposes. Such a signal that produces a non-constant VFDT is a *temporal multifractal* signal. Conversely, should the VFDT be computed from a simple fractal signal, then the resulting VFDT will simply be constant.

The representation of a signal by its VFDT facilitates classification by the complex domain neural network. The VFDT provides a usable set of features upon which classification can be performed because of the dimensionality reduction, underlying feature exemplification, and normalization that occurred as a result of the transformation into the multifractal domain.

2.3 Complex Domain Neural Network

Complex domain three-layer feedforward neural networks (CNN) are used in this paper to perform classification based upon the variance fractal dimensions.

Neural networks that work with real-valued inputs are sufficient for most situations, but when the inputs to the neural network are naturally represented as complex numbers, it is advantageous to use a neural network that takes this representation into account. The fish trajectory signal examined in this paper is a multi-valued signal where each sample consists of three values, one for each of the Cartesian co-ordinate axes. For this particular signal, the majority of the significant information lies along two of the axes. Correspondingly, the signal can be viewed as a complex valued signal whereby samples from the X-axis are used as the real part of the samples in the complex valued signal and samples from the Z-axis are used as the imaginary part. The reason for this representation is that it emphasizes that the trajectory along each of the axes are not independent; rather, they are strongly correlated.

Complex valued data can be provided to a real domain neural network by separating the components of the complex values and providing them separately as inputs; however, any correlation between the components is lost. While in theory, real valued neural networks have the same ability as complex domain neural networks, in practice, the training of complex domain neural networks is typically faster and they often generalize better, especially when only a sparse training set is available.

The architecture of complex domain three-layer feedforward neural networks is similar to their real domain counterparts; the main differences are that each input value and weight is a complex number consisting of both a real and imaginary part. The activation function used in this paper for the neurons is a scaled version of the hyperbolic tangent function, tanh(1.5x) [KaKw92], which is applied to the magnitude of the complex valued input and then multiplied by the unit vector of the input so that the output of the activation function maintains the same direction as the input [Mast94].

This paper uses a single output neuron for each class in order to perform classification. Since the output neurons result in binary decisions for the inclusion or exclusion of an input to a particular class, it is inefficient to employ complex-valued outputs as it does not aid in making the classification decision. Thus, for classification purposes, the imaginary part of the output of these neurons is discarded and the decisions are based solely upon the real part of the output. The network is trained to have the output neuron corresponding to the input vector's class produce a value of 0.9 and with the rest of the output neurons producing -0.9. The classification decision for a given input is then based upon which output neuron produces the highest activation.

The network is restricted to a single hidden layer in this paper since additional layers would increase the complexity of the network as well as the training time substantially, while not adding to the network's ability to classify or generalize significantly because a three-layer neural network is sufficient in nearly all situations. The number of neurons to place in this layer has a dramatic effect on the network's ability to perform the desired operation and the speed at which it executes and trains. The heuristic which specifies that the number of hidden neurons should be set to the geometric mean of the number of input and output neurons is used in this paper.

Through experimentation, it was discovered that this heuristic provides a good balance in that there are enough neurons to learn the desired function without simply memorizing the inputs and the network is restrained to a reasonable size so that the training and execution time of the network is acceptable.

The training of the network is performed using the standard backpropagation algorithm extended to operate with complex values. The partial derivates of the error of the output with respect to the real and imaginary parts of the weights is used as the error gradient to indicate the direction with which to modify the weights. The modifications to the weights in each epoch is given by

$$w_{new_{real}} = w_{old_{real}} - \alpha \frac{\partial \varepsilon}{\partial w_{old_{real}}} \tag{4a}$$

$$w_{new_{imag}} = w_{old_{imag}} - \alpha \frac{\partial \varepsilon}{\partial w_{old_{imag}}} \tag{4b}$$

where ε is the output error, w is the weight in the network, the *real* and *imag* subscripts indicate the real and imaginary parts of the weights, and α is the learning rate. The full derivation of the error gradients shown in (4a) and (4b) can be found in [Mast94].

2.4 Probabilistic Neural Network

An alternative to the complex domain neural network for classification is the probabilistic neural network. The probabilistic neural network (PNN) is an implementation of the Bayes optimal decision rule in the form of a neural network [Spec88]. PNNs have a number of advantages over traditional neural networks in that they tend to train orders of magnitude faster and their classification accuracy asymptotically approaches Bayes optimal. However, PNNs require a comparatively large amount of memory and more time to execute.

2.5 Kohonen Self-Organizing Feature Map

Kohonen self-organizing feature maps (SOFMs) [Koho84] can be used as a clustering algorithm to determine the classes of signals. SOFMs can also be used to perform feature extraction upon the VFDT prior to classification. SOFMs are neural networks that employ unsupervised competitive learning algorithms. These neural networks are referred to as topology-preserving in that the neighbourhood relations of the data are preserved and structure is imposed upon the neurons in the network. This clustering of the data based on their relations allows for the discovery of the underlying structure of the data.

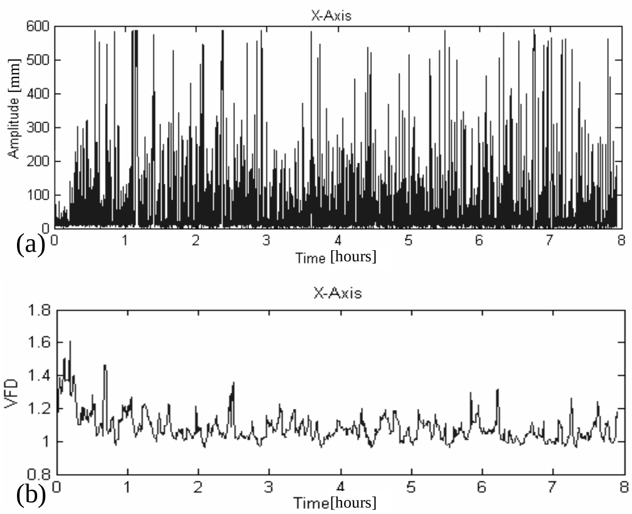

Fig. 2. Fish trajectory signal (a) and its VFDT (b).

3. Experimental Work

3.1 Computational Setup

The dishabituation signals were segmented into lengths of 4096 samples, the longest period of constant behaviour of the fish, and the classes of each of the segments were determined through clustering of the X and Z-axis segments in the time domain with a Kohonen self-organizing feature map [ChCa03]. To train the classification system, a training set of 612 segments, each consisting of 4096 samples, from 9 recordings was used. The testing set applied to the system was made up of 544 segments from 8 recordings that were not used for the training set.

No filtering was performed upon the signals prior to the computation of the VFDTs used to construct the training and testing sets. The VFDTs were computed using a window size of 2048 samples, the largest window in which the fractal dimension of the signals remained constant. This length of 2048 samples, or about 3.4 minutes, as determined by the signal's monofractality can be considered as a weak-stationarity measure. A window displacement of 256 samples was used as it was discovered to give a good resolution of the VFDT while yielding a substantial dimensionality reduction.

Figure 2b shows the VFDT of the signal of Fig. 2a. The first thing to note about the VFDT plot is that the fractal dimensions of the signal changes, indicating that it is multifractal in time. It can further be noted that the samples of the VFDT are normalized dimensions between 1 and 2, which is essential for the classification process. Additionally, the VFDT plots visually seem to correspond to the time domain plots in that they tend to emphasize some of the characteristics in the original signal; the most exemplary characteristic being the initial large changes in the VFDT signals which correspond to the irregular motion of the fish as seen in the time domain plot.

3.2 CNN Experiment

The results of the classification of the input vectors in the testing set using the complex domain neural network are shown in the confusion matrix in Table 1. Overall, the classification system performed well at a correct classification rate of nearly 87%.

Table 1. CNN experiment confusion matrix.

		Experimental				Correct Classification Rate (%)
		1	**2**	**3**	**4**	
Expected	**1**	23	0	0	1	95.83
	2	3	127	8	8	86.99
	3	0	11	151	26	80.32
	4	0	13	3	170	91.40
Average Correct Classification Rate: 86.58%						
95% Confidence Interval: [83.72%, 89.44%]						

The size of each class in the training and testing sets were proportional to their frequency of occurrence in the signals. While the first class had the smallest representation, it was so distinct that all but one of input vectors of this class were correctly classified. Input vectors from the remaining classes were also classified at a high rate, giving confidence to the abilities of the system. As the development of the testing set involved randomness in selecting the input vectors to use for testing, the 95% confidence interval for the classification rate is provided under the confusion matrix in order to bound the true classification rate of the system. The confidence interval was computed by considering each classification of the input vectors in the testing set to be a Bernoulli trial.

3.3 Additional Experiments

Additional experiments were performed using a PNN as the classifier and the results are shown in Table 2. For the first experiment, the X-axis fractal dimensions were used for classification by the PNN. The second experiment was identical to the first, except that the Z-axis signals were used. In both experiments, the results were quite poor. However, by utilizing both the X and Z-axis fractal dimensions for classification, a significantly higher classification rate was achieved.

Table 2. PNN experiments.

Signal	Classification Rate (%)				Average Classification Rate (%)
	1	2	3	4	
X	100	92	59	50	67
Z	63	29	47	91	58
X & Z	100	95	84	95	91

While the results for this last experiment gave slightly higher classification rates than those with the CNN, they are comparable when confidence intervals are taken into account. However, there were some differences in the training and execution times. The PNN trained two orders of magnitude faster than the CNN, while the trained CNN performed classification nearly an order of magnitude faster than the PNN.

These experiments were also repeated using SOFMs to perform feature extraction upon the VFDT prior to classification. For most cases, the classification results when using the SOFMs were slightly lower than when the SOFMs were excluded, but they are essentially equal when confidence intervals are taken into account. Thus, the classification rates remained almost the same despite the fact that fewer features were used for classification.

Further details of these additional experiments can be found in [ChCa03].

4. Conclusions

This work was done to demonstrate the feasibility of classification of self-affine signals by using variance fractal dimensions and complex domain neural networks. This paper has shown that a multifractal characterization of self-affine signals through variance fractal dimensions is an effective means of feature extraction as it provided a sufficient metric upon which to classify the signals used in this paper. Furthermore, the use of complex domain neural networks upon two separate, yet strongly correlated signals were used and shown to be effective in classifying these signals based on its variance fractal dimensions.

Although the classification system created for this paper was shown to perform quite successfully in classifying the non-stationary, self-similar, stochastic, multivariate dishabituation signal, there are a number of extensions that would prove to be valuable in analyzing other signals. First, the classification system could be modified to incorporate hypercomplex input signals involving n-dimensional signals, as there are many examples of multivariate signals composed of more than two significant and correlated components. Second, the fractal dimension trajectory can be generalized to represent both the spatial and temporal multifractal characteristic of a signal through the Rényi fractal dimensions spectrum trajectory. This representation would be valuable for classification, as some signals are not be monofractal in a window, rather they are multifractal in both space and time.

Acknowledgements

Partial financial support for this research was obtained from the Natural Sciences and Engineering Research Council (NSERC) of Canada through a summer scholarship and a grant (W. Kinsner).

References

[ChCa03] V. Cheung and K. Cannons, *Signal Classification through Multifractal Analysis and Neural Networks*. BSc Thesis. Dept. of Electrical and Computer Engineering, University of Manitoba, Winnipeg, MB, 2003, 106 pp.

[KaKw92] B.L. Kalman and S.C. Kwasny, “Why tanh? Choosing a sigmoidal function”, *Proc. Intern. Joint Conf. Neural Networks*, vol, 4, pp. 578-581, June 1992.

[Kins94] W. Kinsner, “Batch and real-time computation of a fractal dimension based on variance of a time series,” *Technical Report*, DEL94-6; UofM; June 15, 1994, 22 pp.

[Koho84] T. Kohonen, *Self-Organization and Associative Memory*. Berlin: Springer-Verlag, 1984, 255 pp.

[Mand82] B. Mandelbrot, *The Fractal Geometry of Nature*. San Francisco, CA: W.H. Freeman, 1982, 468 pp.

[Mast94] T. Masters, *Signal and Image Processing with Neural Networks: A C++ Sourcebook*. New York, NY: Wiley, 1994, 417 pp.

[PeJS92] H.O. Peitgen, H. Jürgens, and D. Saupe, *Chaos and Fractals: New Frontiers of Science*. New York, NY: Springer Verlag, 1992, 984 pp.

[Spec88] D.F. Specht, “Probabilistic neural networks for classification, mapping, or associative memory”, *Proc. IEEE Intern. Conf. Neural Networks*, vol. 1, pp. 525-532, July 1988.