



Identification of spectral lines of elements using artificial neural networks

M. Saritha*, V.P.N. Nampoori

International School of Photonics, Cochin University of Science and Technology, Cochin University P O, Cochin, Kerala, PIN 682 022, India

ARTICLE INFO

Article history:

Received 27 May 2008

Received in revised form 25 October 2008

Accepted 25 October 2008

Available online 11 November 2008

Keywords:

Identification

Neural network applications

Spectral analysis

Spectroscopy

ABSTRACT

Artificial neural networks (ANNs) are relatively new computational tools that have found extensive utilization in solving many complex real-world problems. This paper describes how an ANN can be used to identify the spectral lines of elements. The spectral lines of Cadmium (Cd), Calcium (Ca), Iron (Fe), Lithium (Li), Mercury (Hg), Potassium (K) and Strontium (Sr) in the visible range are chosen for the investigation. One of the unique features of this technique is that it uses the whole spectrum in the visible range instead of individual spectral lines. The spectrum of a sample taken with a spectrometer contains both original peaks and spurious peaks. It is a tedious task to identify these peaks to determine the elements present in the sample. ANNs capability of retrieving original data from noisy spectrum is also explored in this paper. The importance of the need of sufficient data for training ANNs to get accurate results is also emphasized. Two networks are examined: one trained in all spectral lines and other with the persistent lines only. The network trained in all spectral lines is found to be superior in analyzing the spectrum even in a noisy environment.

© 2008 Elsevier B.V. All rights reserved.

1. Introduction

Spectrum of various compounds and elements are taken up for spectroscopic studies. In spectroscopic studies, the spectrum of the sample, taken using a spectrometer, is plotted on a graph and the various photo-peaks are identified. The spectrum of a sample contains the characteristic spectral lines of all the elements present in the sample. Thus, it is a linear superposition of the spectral lines of the elements present, but scaled. Even the weak spectral line of a particular element is obtained if the concentration of that element in the sample is high. Also, the strongest line of an element becomes unobservable if its concentration is very low. Under such conditions, only persistent lines are obtained. Hence a spectrum is a linear superposition of all the weak, strong and persistent lines of all the elements present in the sample. Usually a spectrum contains spurious peaks over and above the original peaks. The photo-peaks obtained for the sample help to identify the elements present in the sample and also to test the purity of the elements. In this paper, the possibility of using ANNs to identify the spectral lines of the elements even in the presence of spurious signals has been explored.

Keller and Kouzes have shown that Gamma spectral analysis can be successfully done using ANNs [6]. Also the same team has done an identification of the nuclear spectrum for waste water handling [7]. Olmos has also suggested an automation analysis of radiation spectrum using ANNs [4] and [5]. All these researchers have suggested that trained ANNs can be used for automation of specific types of spectrometers. Keller and

Kouzes used data generated by Monte Carlo simulations and automated the spectrometer. The input to the ANN is provided by the different channels of the spectrometer without giving any specifications to the wavelength of the obtained spectrum. Here, an attempt is done to take into consideration the characteristic spectral lines of elements and their wavelength and intensity in the whole visible range. The spectral lines in the visible range of Cadmium, Calcium, Iron, Lithium, Mercury, Potassium and Strontium are chosen for the studies. Also the performance of the system with respect to intensity variations and different noise levels are evaluated. This technique can be used with any type of spectrometer.

2. Artificial neural networks

ANNs are used in a wide variety of data processing applications where real-time data analysis and information extraction are required. One advantage of the neural network approach is that most of the intense computation takes place during the training process. Once the ANN is trained for a particular task, operation is relatively fast and unknown samples can be identified [6]. Here an automatic approach for peak identification is discussed using artificial neural networks (ANN).

Work on artificial neural networks (ANN) has been motivated right from its inception because of the understanding that human brain works in an entirely different way from conventional digital computer. The brain is a highly complex, nonlinear and parallel information processing system. It has the capability to organize its structural constituents, known as neurons, so as to perform certain computations many times faster than the fastest digital computer in existence today.

An ANN is a massive parallel distributed processor made up of simple processing units which has a natural propensity for storing

* Corresponding author. Tel.: +91 9447608163; fax: +91 484 2576714.

E-mail addresses: saritha_madhu@rediffmail.com (M. Saritha), nampoori@gmail.com (V.P.N. Nampoori).

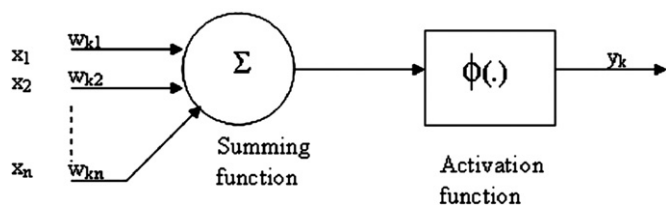


Fig. 1. A neuron model.

experimental knowledge and making it available for use. It resembles brain in two aspects: Knowledge is acquired by the network from its environment through a learning process. Interneuron connections, known as synaptic weights, are used to store the acquired knowledge. The procedure used to perform the learning process is called a learning algorithm, the function of which is to modify the synaptic weights of the network in an orderly fashion to attain a desired design objective.

A neuron, shown in Fig. 1, is an information processing unit that is fundamental to the operation of a neural network. Each interconnected node, or neuron, is defined as a simple processing element whose output y is given by

$$y_k = \Phi \left(\sum_{j=1}^n w_{kj} x_j \right) \quad (1)$$

where x_1, x_2, \dots, x_n are the inputs to the neuron, $w_{k1}, w_{k2}, \dots, w_{kn}$ are the synaptic weights, and $\Phi(\cdot)$ is known as an activation function. The activation function is the key to the behavior and performance of the network. The activation function is for limiting the amplitude of the output of the neuron. The activation function is also referred to as a squashing function as it squashes the permissible amplitude range of the output signal to a finite value.

The purpose of the learning algorithms is to find the synaptic weight w_{kj} suitable to handle the given problem. There are two fundamental learning paradigms: learning with a teacher and learning without a teacher. Learning with a teacher is known as supervised

learning and the other unsupervised learning. Here, a feedforward ANN with a supervised learning is discussed. In supervised learning, the system is provided with a target output, T . The aim of the learning algorithm is to adjust the synaptic weights in such a way as to minimize the error between the output of the network Y and target output, T .

Problem specifications help to define the network in the following ways: the number of neuron in the input layer is the same as the number of inputs for the problem, the number of neurons in the output layer is determined by the required number of outputs for the problem and the choice of the activation function is partly determined by problem specification of the output. If the response of the system is linear usually linear activation functions are selected. Otherwise non linear functions are taken [2] and [3].

3. Modelling issues

The development of a successful ANN project constitutes a cycle of six phases [1]. The first phase is the problem definition and formulation. In the present case, the problem is to identify the spectral lines in of seven elements namely Cadmium, Calcium, Iron, Lithium, Mercury, Potassium and Strontium in the visible range. Second is the system design phase. Usually supervised learning is suitable. Analyzing the spectrum of elements taken, from data hand-book, it is evident that the spectral lines occur in discrete values at different wavelengths as in Fig. 2. These consist of strong, persistent and weak lines.

In spectroscopic observations made with low concentrations of a particular element relative to the other elements in the sample, the number of observable lines of the element is found to decrease with decreasing concentration until only the most “persistent” or “sensitive” lines remain. Some authors refer to these lines as the ultimate lines. Although the ultimate lines depend in principle on the sample, the spectrometer, and other features of the experiment, a relatively small group of lines can be specified for each element that will include the ultimate lines as observed over a broad range of experimental conditions. These lines are designated as “persistent lines” [8].

Spectrum require for the investigation is recorded using a CCD camera. They consist of various photo-peaks which are characteristic

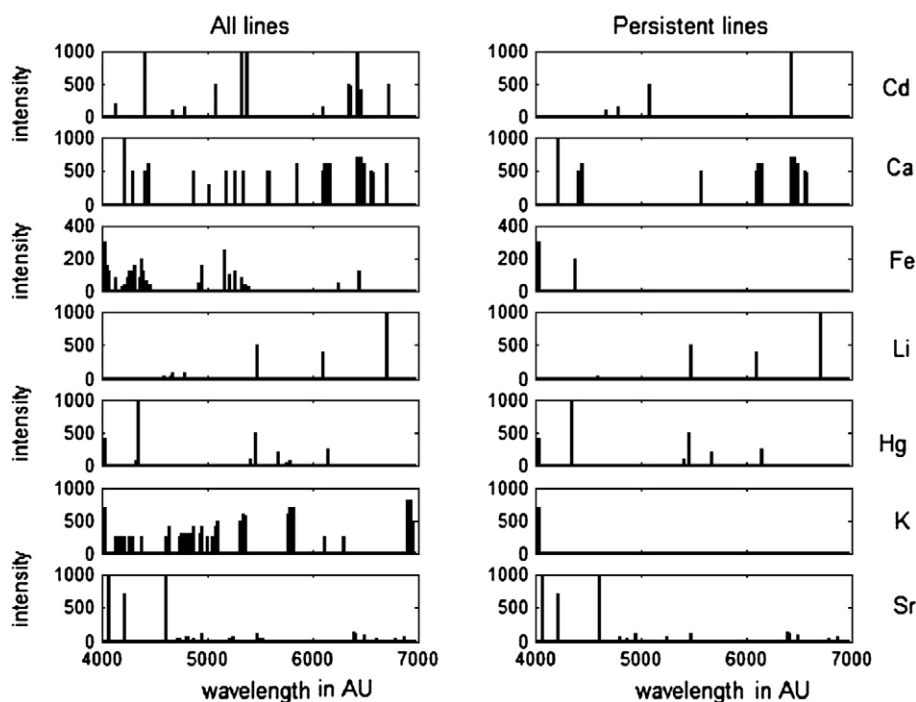


Fig. 2. Spectral lines of elements with (a) all lines and (b) persistent lines.

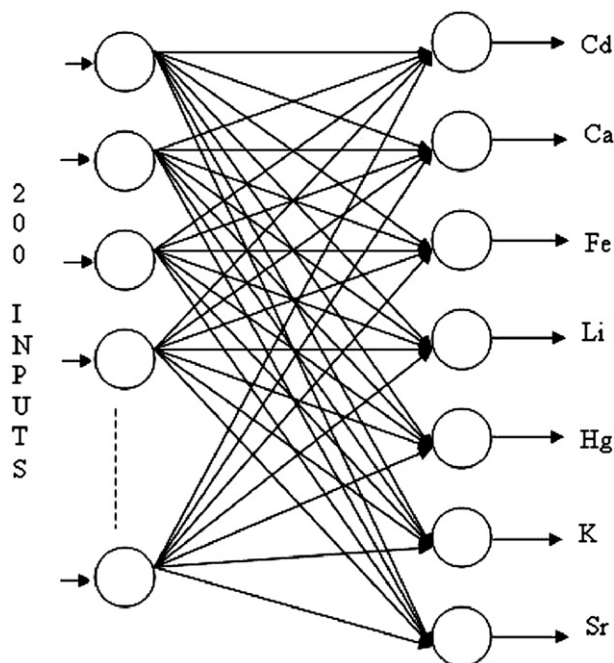


Fig. 3. An ANN to identify the elements.

spectral lines of the elements which constitute the sample. Also it is not necessary that all characteristic lines of each constituent element be found in the spectrum. But the probability of occurrence of the persistent lines of the elements is very high. Thus the spectrum taken is a linear superposition of the spectra of the constituent elements in the sample. Indeed, the photo-peaks do not have the same relative intensity as specified in the data handbook. They are scaled. So, if e_i represents the spectrum of element i in the sample, then the intensity of the characteristic line of the sample S can be given as:

$$S = \sum_i \alpha_i e_i \quad (2)$$

where α_i is the scaling factor of the relative intensity of the spectral lines of element i .

The output has a linear response with the input. Therefore, the classification system should have a linear response with respect to the input. An ANN designed to have a linear response employs linear activation functions. A feedforward ANN that implements linear activation function can be reduced to a network with a single input layer and a single output layer. The ANN used in the present application has a single input and a single output layer as illustrated in Fig. 3. This can be trained using linear perceptron models or using optimal linear associative memory (OLAM) algorithms. A linear perceptron does not converge to accurate results and OLAM is most suited for such applications as shown by Keller and Kouzes [6].

The optimal linear associative memory (OLAM) approach is based on a simple matrix associative memory model. It was developed in the early 1970s as a content addressable memory and is useful in situations where the input consists of linear combinations of known patterns. It is

Table 1
The characteristic spectral lines for cadmium in the range 400–500 nm

Relative intensity	Wavelength in Å
200	4134.768
1000	4415.63
100	4678.149
150	4799.912

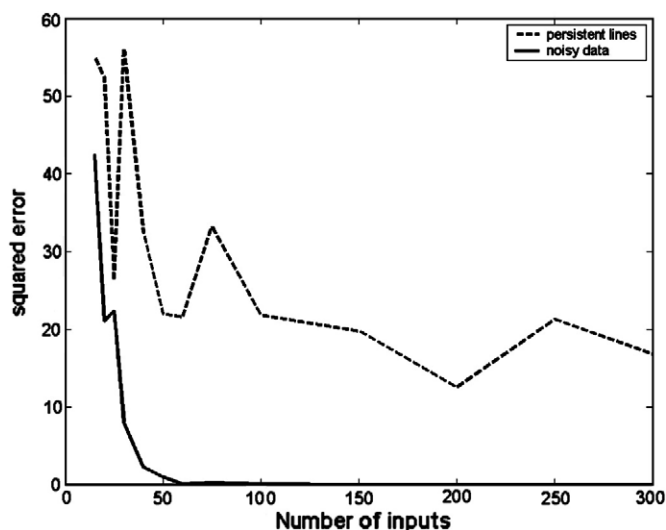


Fig. 4. Plot to find the number of inputs.

an improvement over the original matrix memory approach in that it projects an input pattern onto a set of orthogonal vectors where each orthogonal vector represents a unique pattern. With linear activation functions, the training is a straight forward matrix orthogonalization process where each pattern from the training set is made to project onto a separate, unique orthogonal axis in the output space [6].

3.1. OLAM weight specification

Step 1. Form matrices of spectra. Arrange spectra as columns in an $n \times p$ dimensional matrix \underline{X} , where n is the number of inputs and p is the number of elements and target as columns in a $p \times p$ dimensional matrix \underline{T} .

Step 2. Generate inverse of the spectral matrix \underline{X} . Since \underline{X} is generally not a square matrix, a pseudo-inverse technique is used to generate \underline{X}^\dagger .

(\dagger indicates pseudo-inverse)

Step 3. Form the synaptic weight matrix.

$$\underline{W} = \underline{TX}^\dagger$$

The third phase of an ANN development project is system realization. The spectral lines data from the handbook for each element is as shown in Fig. 2. In the system realization phase, the number of input neurons and the number of output neurons are to be determined. The problem specification determines both. Usually the number of output neurons is taken as the number of outputs required for the problem. Since there are seven elements to be identified in this study, the number of output neurons is taken as seven. Next phase is to determine the number of input neurons. In this particular study the number of input neurons is determined by training and testing. Two sets of data are given for testing: One, a set of noisy data and other the persistent lines. As the probability of occurrence of persistent lines (ultimate lines) is the highest, they should be identified in any worse condition, even though they are few in number.

The spectral data is scanned with a resolution of 1 Å. This is to ensure discretion with spectral lines which are very close to each other. In the nanometer scale, they are treated as the same line. There are about 3000 wavelength points with their intensities. As seen in Fig. 2, most of these points have intensity values of zeroes. To be more precise, consider the element Cadmium with its characteristic spectral lines in the range 400–500 nm, given by Table 1.

When scanned with a resolution of 1 Å, up to a wavelength 4134 Å, the intensity value is zero and at 4135 Å it is 200, then up to 4415 Å it is

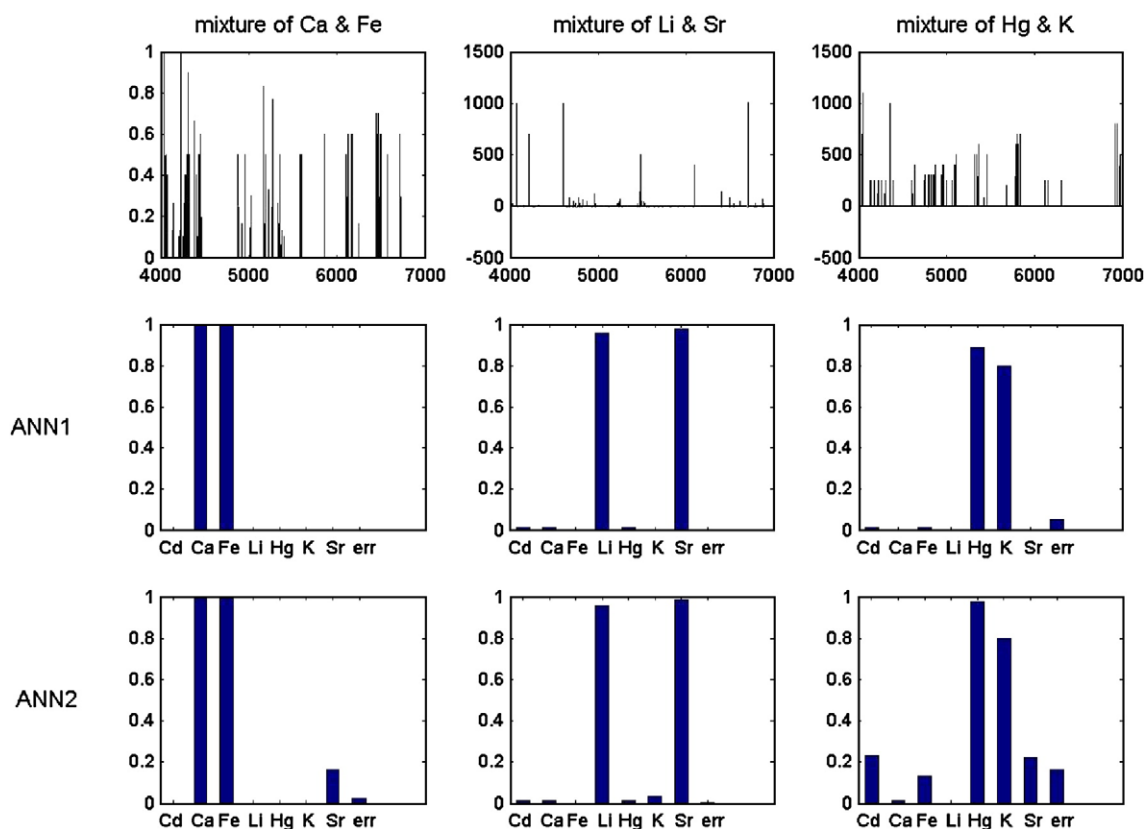


Fig. 5. Output of the ANN for different samples.

again zero and at 4416 Å it is 1000 and so on. When most of the data contains very low values or zeroes, the learning algorithms will not converge to accurate results. So a reduction in data is required. The most common method in data reduction is to find the area under the curve formed by the data points which form a polygon. It is to determine the optimum number of wavelength points that is required to make the polygon so as to get a better result from the trained ANN. The data is divided into equal parts and the area is taken for each segment. As an example, consider that the data is segmented into 150 equal parts each of 20 wavelength points and their intensities. Area is taken by considering the polygon with these 20 points. Therefore the data is now reduced from 3000 data points to 150 data points. The data is then normalized, so that there are now 150 input nodes with normalized data. This kind of data reduction is done for each element and is arranged in a matrix form. The matrix, \underline{X} (as in the OLAM algorithm), is now having 150 rows and 7 columns, each column specifying an element. Since 7 elements are to be identified, the target matrix \underline{T} is a 7×7 matrix. By taking the pseudo inverse of \underline{X} , it becomes a 7×150 matrix. The weight matrix, 7×150 , is calculated as per the OLAM algorithm given. Testing of the result is done with the persistent line data and the noisy data. For the persistent line and noisy data of each element, the data is again scanned with 1 resolution and is segmented into 150 equal parts, area is evaluated and is normalized. The output is verified for these inputs as per Eq. (1) and the error is calculated. This is done for segments of varied lengths. For noisy data, the error goes on decreasing as the number of input nodes increases. But when the number of input segments is 200, the system has minimum error for the identification of persistent lines as shown in Fig. 4. Therefore, the number of inputs to the system is 200. Thus the network is ready for training.

The goal of the training is to learn an association between the spectra and the labels representing the spectra. The training process for the OLAM is a non-iterative process and it converges very fast. The

weight matrix is obtained using pseudo-inverse rule. Two types of ANNs are trained, one with all the spectral lines (ANN1) and the latter with the persistent lines (ANN2) alone. Only the visible range (400–700 nm) of the spectrum is considered in our studies. The persistent lines are very few in number. For elements like potassium there are only two persistent lines in the required range, as shown in Fig. 2, whereas, there are about 44 spectral lines for potassium in the

Table 2
Output obtained for the various samples by the two ANNs

	Cd	Ca	Fe	Li	Hg	K	Sr	Error
A mixture of Fe and Hg								
ANN1	0	0	1	0	1	0	0	0
ANN2	0	0.01	1	0	1	0	0.03	0.001
A mixture of Hg and Sr								
ANN1	0	0	0.01	0	0.99	0	0.99	0.0003
ANN2	0	0	0	0	1	0.02	0.99	0.0005
A mixture of Cd and Sr								
ANN1	1	0	0.01	0	0.01	0	1	0.0002
ANN2	0.99	0.02	0	0	0.01	0.03	1	0.0015
A mixture of Ca and Li								
ANN1	0	1	0	1	0.01	0	0.01	0.0002
ANN2	0.01	0.99	0.01	1	0.01	0	0.13	0.0173
A mixture of Li and K								
ANN1	0.01	0	0	1	0	0.8	0.02	0.0405
ANN2	0.23	0.01	0.11	0.99	0	0.8	0.23	0.1581
A mixture of Ca (peak reduced to 80%) and Hg (peak reduced to 70%)								
ANN1	0	0.8	0	0	0.69	0	0.01	0.0002
ANN2	0.01	0.79	0	0	0.73	0	0.11	0.0132

Each column represents different elements. RMS error is given in the right hand column.

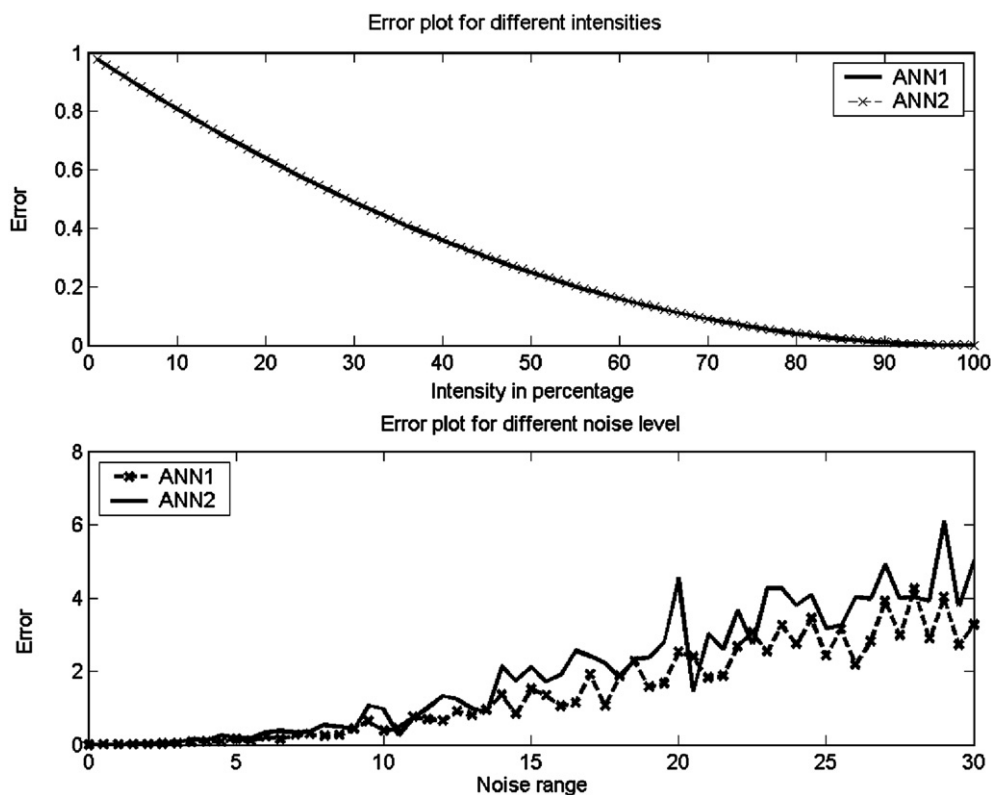


Fig. 6. Root mean square error plots for (a) different intensities (b) noise level.

visible range. The ANNs are tested with known samples and unknown samples.

4. Results and discussions

Now the system is to be tested. Spectra of mixtures were generated by combining spectra of different elements. Random noise is also added to the spectra of the mixture. The data is scanned with a resolution 1 and is segmented into 200 equal parts and the area is evaluated. The data thus got is normalized and fed to the system. The output obtained for each ANN for 3 different mixtures is as shown in Fig. 5. The first sample is a mixture of calcium and iron in pure form without any noise. But the other two samples, one a mixture of lithium and strontium and other a combination of mercury and potassium, are noisy data. ANN1 gives a consistent performance than ANN2 even in noisy environment. This is because the number of observable spectral lines in the visible range is very high compared to the persistent lines. For the third sample which is a mixture of Hg and K, ANN1 gives more accurate result than ANN2. For K, there are only two persistent lines in the visible range and these lines are very close to each other also. To identify K with ANN2 is a very tedious task and most of the time it leads to errors. ANN1 on the other hand gives a very consistent result. In this context, the need for sufficient spectral lines in the required range for training is emphasized.

More results are shown in Table 2 also. The identification of the spectral lines of Fe also gave some errors even with ANN1. From Fig. 2, the highest relative intensity of Fe is only 400 when compared to other elements having highest value of 1000. In the training phase, since the data is normalized, Fe requires no enhancement. But when the spectrum of Fe is combined with that of others, the intensity of the spectral lines of Fe becomes very low. So the spectrum of Fe is enhanced before combining with that of others.

ANN1 correctly identifies most of the spectral lines of elements fed to it. But ANN2 had hard times in differentiating potassium with strontium. In certain cases, ANN1 shows presence of mercury, which is not present. Hg has only 15 spectral lines in the visible range and most

of them have very low intensity. Certain spectral lines of Hg coincide with the spectral lines of elements like K. However, the errors with ANN1 were always smaller than that with ANN2.

The performance of the networks with varying relative intensities and noise levels was carried out. First the intensities of the lines were reduced. Here, no noise was added and all the spectral lines in both data sets were considered but with reduced intensity. With the intensity as the original, the outputs of the ANNs were 1. When the intensity was reduced, the output also correspondingly reduced. As shown in Table 1, when the intensity of the spectral lines of Ca was reduced to 80%, ANNs output was only 0.8. The error plot for the output obtained for different intensity levels are shown in Fig. 6(a). It is seen that the performance of both the networks are the same when the intensity is reduced. When the intensity is reduced below 70% of the original relative intensity value, the network gives errors. Here, it is worthwhile to note that all spectral lines in the visible range are considered.

The networks are now tested with different noise levels. The output for different noise levels are shown in Fig. 6(b). Random noise is added to the data at different noise levels. The graph shows the average error for 1000 such data. Here the performance of ANN1 is better than ANN2. This can be seen in Fig. 5 also. When the noise levels are very low, the network output is not affected. But as the noise levels are increased, the output of the network shows errors. As shown in the Fig. 6(b), noise levels cannot be increased beyond a factor of 7 for both ANNs. Noise levels in practical cases will not be very high. From this it is clear that random noise with normal distribution will not affect the performance of the network. Only if the noise amplitude is increased to 7 times its original value, some error occurs, which is not a practical case. Thus ANN1 is preferred over ANN2.

5. Conclusion

The initial results of our research have demonstrated the pattern recognition capabilities of the neural networks. It has also emphasized the need for a large number of spectral lines in the desired range for

the accurate classification of elements. ANN1, which is trained with more number of spectral lines than ANN2, gives a better performance. This is because ANNs can easily generalize when data is large. The classification is attributed to the orthogonalization process used by the OLAM during training. Since this training is a non iterative process, the OLAM offers a substantially shorter training time. One of the disadvantages of the OLAM, is that all the spectral lines of each element – weak, strong and persistent within the visible range – are used for training. Good results are obtained when all the lines are considered. But in a practical case, it is not possible to obtain all the spectral lines in a given range. Further work has already been initiated to train a network with the characteristic lines of the elements and to observe the performance of the network in practical situations.

Acknowledgement

The first author would like to acknowledge Prof. P Radhakrishnan of the International School of Photonics, Cochin University of Science and Technology for his timely help extended to her.

References

- [1] I.A. Basheer, M. Hajmeer, Artificial neural networks: fundamentals, computing, design and application, *Journal of microbiological methods* 43 (2000) 3–31.
- [2] T. Kohonen, *Self Organization and Associative Memory*, third ed. Springer-Verlag, New York, 1989.
- [3] Martin T. Hagan, Howard B. Demuth, Mark Beale, *Neural Network Design*, first ed. Thomson Learning, Boston, 1996.
- [4] P. Olmos, J.C. Diaz, J.M. Perez, P. Aguayo, P. Gomez, V. Rodellar, Drift problems in the automatic analysis of gamma ray spectra using associative memory algorithms, *IEEE Transactions on Nuclear Science* 41 (1994) 637–641.
- [5] P. Olmos, J.C. Diaz, J.M. Perez, P. Gomez, V. Rodellar, P. Aguayo, A. Bru, G. Garcia-Belmonte, J.L. de Pablos, A new approach to automatic radiation spectrum analysis, *IEEE Transactions on Nuclear Science* 38 (1991) 971–975.
- [6] Paul E. Keller, Richard T. Kouzes, Gamma spectral analysis via neural networks, *IEEE Transactions on Nuclear Science* (1995) 341–345.
- [7] Paul E. Keller, Lars J. Kangas, Gary L. Troyer, Sherif Hashem, Richard T. Kouzes, Nuclear spectral analysis via artificial neural networks for waste handling, *IEEE Transactions on Nuclear Science* 42 (1995) 709–715.
- [8] J.E. Sansonetti, W.C. Martin, Handbook of basic atomic spectroscopic data, *Journal of Physical and Chemical Reference Data* 34 (4) (2005) 1559–2259.