A Framework of Hyperspectral Image Compression using Neural Networks

Yahya M. Masalmah, Ph.D[1], Christian Martínez-Nieves[1], Rafael Rivera-Soto[1], Carlos Velez[1], and Jenipher Gonzalez[1]
[1] Universidad del Turabo, Puerto Rico, yamasalmah@suagm.edu,clmartinez151@gmail.com, rafaelriverasoto@gmail.com, cronos_80@hotmail.com, jgonzalez641@email.suagm.edu

Abstract*– Hyperspectral image analysis has gained great attention due to its wide range of applications. Hyperspectral images provide a vast amount of information about underlying objects in an image by using a large range of the electromagnetic spectrum for each pixel. However, since the same image is taken multiple times using distinct electromagnetic bands, the size of such images tend to be significant, which leads to greater processing requirements. The aim of this paper is to present a proposed framework for image compression and to study the possible effects of spatial compression on quality of unmixing results. Image compression allows us to reduce the dimensionality of an image while still preserving most of the original information, which could lead to faster image processing. This paper presents preliminary results of different training techniques used in Artificial Neural Network (ANN) based compression algorithm.

***Keywords*— Image Compression, ANN, HSI**

Digital Object Identifier (DOI): http://dx.doi.org/10.18687/LACCEI2015.1.1.189
ISBN: 13 978-0-9822896-8-6
ISSN: 2414-6668

A Framework of Hyperspectral Image Compression using Neural Networks

Yahya M. Masalmah, Ph.D[1], Christian Martínez-Nieves[1], Rafael Rivera-Soto[1], Carlos Velez[1], and Jenipher Gonzalez[1]

[1]Universidad del Turabo, Puerto Rico, yamasalmah@suagm.edu,clmartinez151@gmail.com, rafaelriverasoto@gmail.com, cronos_80@hotmail.com, jgonzalez641@email.suagm.edu

Abstract– Hyperspectral image analysis has gained great attention due to its wide range of applications. Hyperspectral images provide a vast amount of information about underlying objects in an image by using a large range of the electromagnetic spectrum for each pixel. However, since the same image is taken multiple times using distinct electromagnetic bands, the size of such images tend to be significant, which leads to greater processing requirements. The aim of this paper is to present a proposed framework for image compression and to study the possible effects of spatial compression on quality of unmixing results. Image compression allows us to reduce the dimensionality of an image while still preserving most of the original information, which could lead to faster image processing. This paper presents preliminary results of different training techniques used in Artificial Neural Network (ANN) based compression algorithm.

Keywords—Image Compression, ANN, HSI

I. INTRODUCTION

With more advancements in high performance computing, hyperspectral image analysis has become an area of interest to many. According to [1], hyperspectral image analysis (HIA) consists in collecting hundreds or even thousands of measurements (at multiple wavelength channels) for the same area of any surface particular surface of interest. From this, a resulting data cube is created, which is a stack of images taken with different electromagnetic bands, where each pixel (vector) has an associated spectral signature. This spectral signature is a sort of 'fingerprint' that identifies the underlying object in the pixel. Since such images are usually taken using hundreds, or even thousands of spectral bands, the images tend to consist of several Gigabytes of data. Although many high-performance computing techniques have been employed to deal with the increasing computational costs involved with the analysis of hyperspectral imagery, image compression techniques can also be employed into the analysis process to improve overall performance.

Since hyperspectral images (HSI) can be viewed as just an array of images, which are in turn 2-dimensional matrices, distinct image compression techniques can be applied to them. As mentioned in [2], image compression consists in taking image data as input, and reducing its dimensionality as much as possible, while still retaining the original data when decompressed. Artificial neural network based techniques have resulted very useful when it comes to image compression due to their noise reduction and learning capabilities.

In applying image compression to hyperspectral imagery [2, 3, 4, 5], it's believed that the computational cost of the analysis process can be significantly reduced. Although the dimensionality of the HSI would be reduced before it is analyzed, it is expected that the results from the compressed analysis be fairly similar to those of the normal HSI analysis process.

II. IMAGE COMPRESSION

Several types of artificial neural networks (ANN) based image compression models have been employed in the past. Such models can be categorized as either linear or non-linear neural networks, according to their activation function. Although some previously developed compression algorithms rely on modular structured neural networks, as well as self-organizing neural networks, a multi-layered artificial neural network based algorithm has been employed for the compression process. Such algorithm uses a feed-forward-back propagation approach, consisting of two layers, to successfully achieve compression [2].

On the other hand, the compression algorithm will also make use of a learning rule to train the neural network. The network would be created based on an input layer, a hidden layer, and an output layer. The image would thus be divided into sub-blocks, where the pixel's grey-level values would be reshaped into a column vector, and be used as input for the neural network through the input layer.

13th LACCEI Annual International Conference: "Engineering Education Facing the Grand Challenges, What Are We Doing?"
July 29-31, 2015, Santo Domingo, Dominican Republic **ISBN:** 13 978-0-9822896-8-6 **ISSN:** 2414-6668
DOI: http://dx.doi.org/10.18687/LACCEI2015.1.1.189

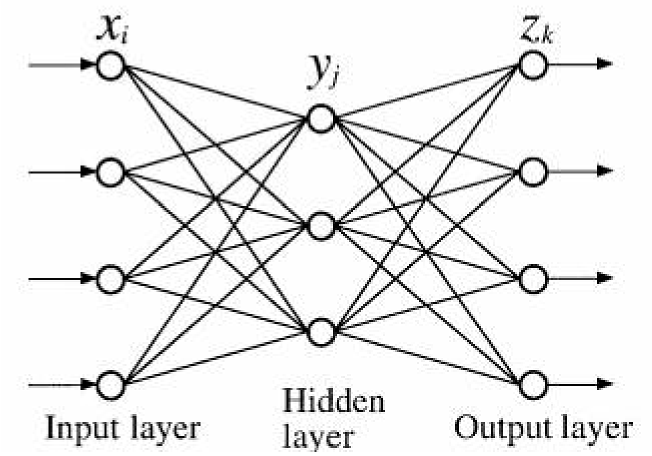

Fig. 1 Example of feed-forward-back propagation neural network

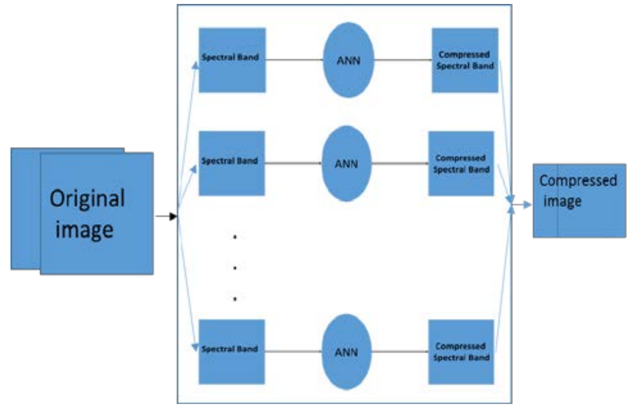

Fig. 3 Proposed compression Algorithm of Hypercube

As illustrated in Fig.1, both the input and output layers are connected to all neurons within the hidden layer, while the hidden layer always contains less neurons than the previous two. Assigning less neurons in the hidden layer is key to achieving true compression. Each layer represents the picture in different states. The input layer is the original picture, hidden layer is the compressed picture, and the output layer is the decompressed picture. For the purpose of HIS, the same process can be viewed as depicted in Fig. 2.

III. Proposed Compression Framework

The proposed compression methodology is a structure which deals with each spectral band as individual image as shown in Fig. 3. This structure makes it suitable for parallel processing which in turns minimizes the processing time. The algorithm will repeatedly read one spectral band, compress it, and stack the compressed band into a lower dimension hypercube. The developed algorithm possess a time complexity of O(nmp) where, n, m, and p are number of rows, number of columns, and number of spectral bands, respectively.

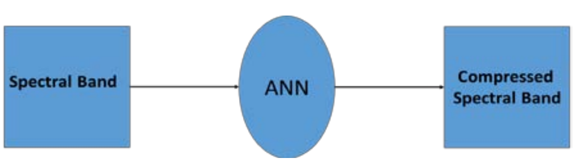

Fig. 2 Example of HSI spectral band compression process

A. Algorithm

The following steps provide a quick summary of the algorithm used to implement the ANN based image compression per spectral band:

1. Divide the original image into sub blocks.
2. Arrange the column vectors in a matrix.
3. Simulate the network.
4. Obtain the output matrices of the hidden layer and output layer.
5. Post process them to obtain the compressed image.

Several Neural network training methods were tested and results are presented in Table I.

IV. HSI Analysis with Compression

Having chosen a compression algorithm, a test HSI image could be analysed, both with and without compression to establish just how reliable the compressed results actually are. For the HSI analysis process, two algorithms, denoted as PMF2 and NNSTO, implemented in MATLAB are used. It’s important to mention that the Quasi-Newton compression algorithm is also implemented in MATLAB.

When reading hyperspectral images into MATLAB, the images are formatted as two-dimensional images (although in fact it’s actually an array of images at different wave lengths). This means that no further preparations have to be done to the image of interest before compression.

The HSI analysis process was realized twice: without compression, and with compression. When not compressing the image, the image was simply loaded into the MATLAB environment and then analysed using the PMF2 algorithm. On the other hand, in the compressed testing, the image has to be compressed using the Quasi-Newton (BFG) algorithm, and the resulting compressed image would be analysed with the PMF2 algorithm.

V. RESULTS

To validate the proposed algorithm, a testing image was used. The algorithm was executed using different Neural Network training techniques. The used training techniques were compared based on the performance amount, and the number of Epochs to get to the performance goal. The performance final values and the number of Epochs are shown in Table 1.

TABLE I
TRAINIGN TECHNIQUES: PERFORMANCE AND EPOCHS RESULTS

Training Technique	Performance	No. Epochs
Trainbfg	0.00087	572
Traincgf	0.00103	839
Traincgp	0.00079	1001
Traingd	0.21846	1001
Traingda	0.02896	141
Trainlm	0.00066	96
Trainoss	0.00119	893

The performance plot versu the number of Epochs needed to reach the performance goal for Quasi-Newton(BFG), Conjugate-Gradient(CGF), Conjugate-Gradient (CGP), Gardient Descent(GD), Gradient-Descent(GDA), Levenberg-Marquadt(LM), and Quasi-Newton (OSS) are shown in Fig. 4 through Fig. 10, respectively. Some training techniques looks better that the Quasi-Newton in terms of performance and number of Epochs but they are bad at memory management.

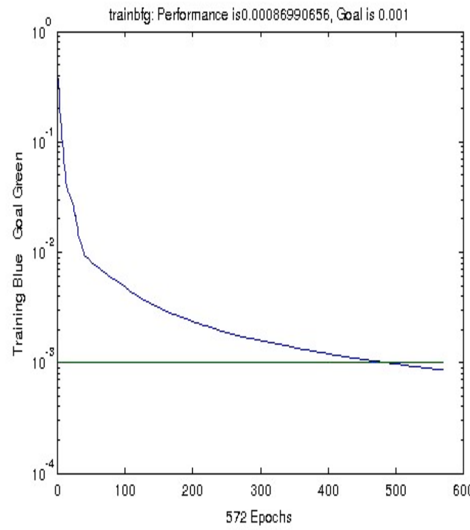

Fig. 4 Performance vs Epoch (trainbfg)

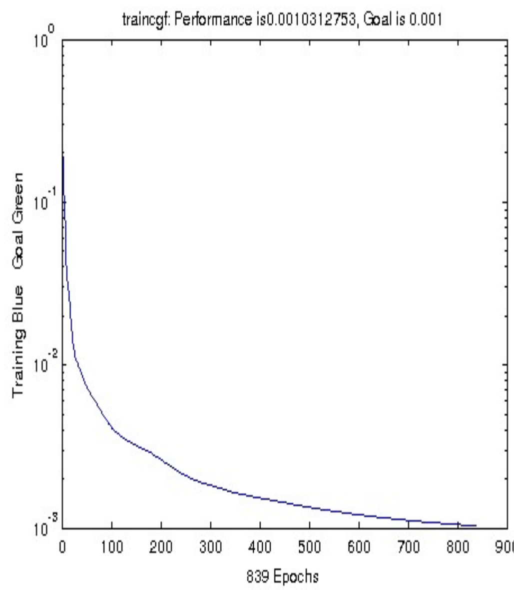

Fig. 5 Performance vs Epoch (traincgf)

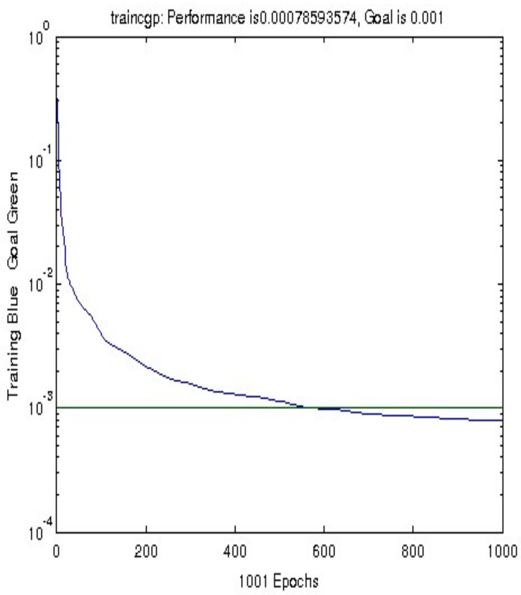

Fig. 6 Performance vs Epoch (traincgp)

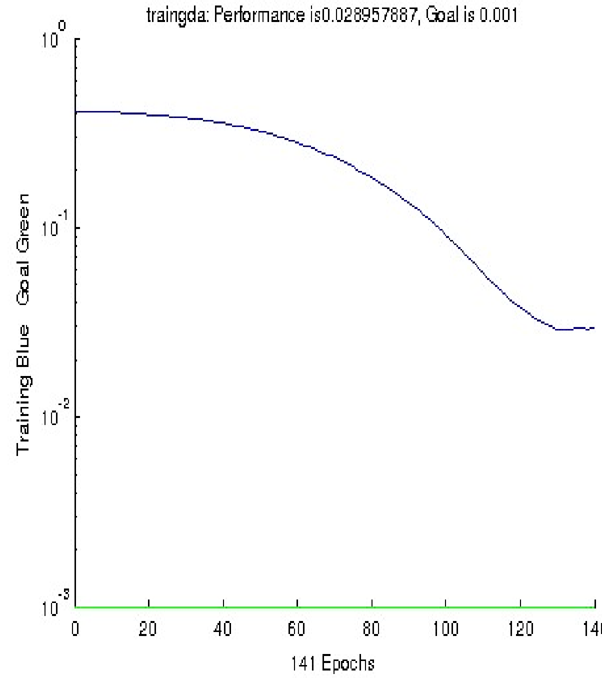

Fig. 8 Proposed compression Algorithm of Hypercube

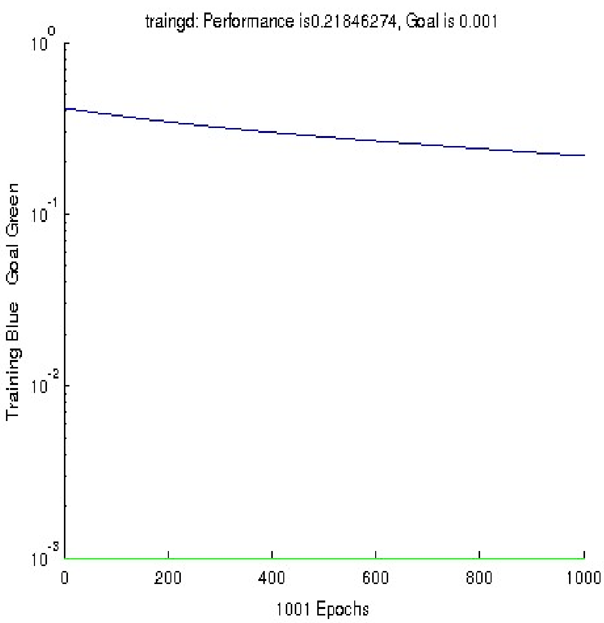

Fig. 7 Performance vs Epoch (traingd)

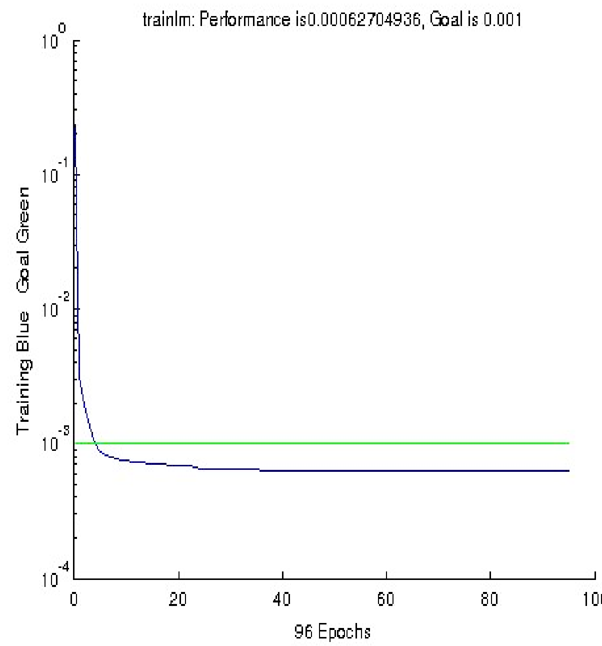

Fig. 9 Performance vs Epoch (trainlm)

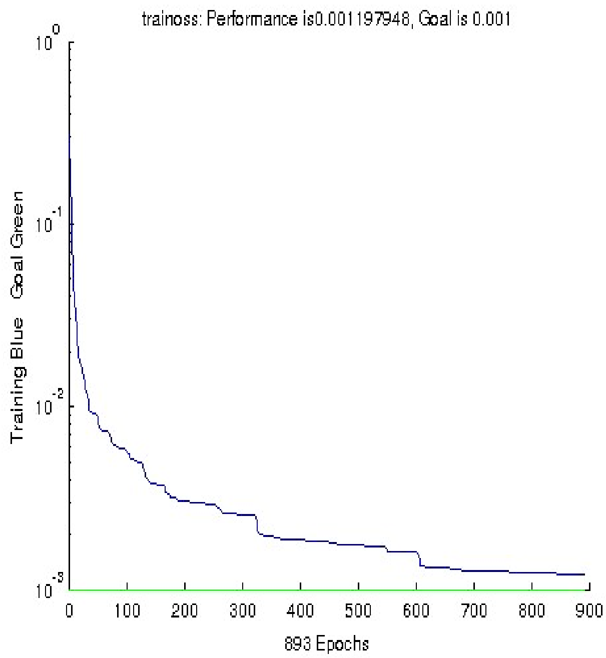

Fig. 10 Performance vs Epoch (trainoss)

VI. CONCLUSIONS AND FUTURE WORK

This paper proposed a framework for hyperspectral image compression. The proposed framework focuses on repeatedly call the Artificial Neural Network algorithm compression algorithm for each spectral band individually. Different training algorithms were used. A comparison between the used training algorithms based on their performance and number of Epochs to achieve the desired performance goal. Results show that Qausi-newton algorithms are suitable for the compression due to their acceptable performance.

For future work, we will study the effect of the image compression on the quality of results obtained from unmixing of hyperspectral images. Our results show significant size reduction for hyperspectral images. This indeed will impact the execution time.

ACKNOWLEDGMENT

This research was partially supported by Department of Energy under Dr. Samuel Massie Chair of Excellence grant.

REFERENCES

[1] A. Plaza, Q. Du, Y. Chang, and R. King. High Performance Computing for Hyperspectral Remote Sensing. IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING, VOL. 4, NO. 3, SEPTEMBER 2011.

[2] E. Watanabe, and Katsumi, Mori, "Lossy Image compression using a Modular Structured Neural Network," Proc. of IEEE signal processing society workshop, pp.403-412, and 2001.

[3] Q. Du, and J. Fowler, "Hyperspectral Image Compression Using JPEG2000 and Principal Component Analysis," IEEE Geoscience and Remote Sensing Letters, vol. 4, pp.201-205, April 2007.

[4] Q. ul Haq, L. Shi, et. al, "A Robust Band Compression Technique for Hyperspectral Image Classification," *Intelligent Computing and Intelligent Systems(ICIS)*, vol. 2, no. 2, pp. 196-200, October 2010.

[5] H. Charif, and F. Salam, "Neural Network-based Image Compression System," Proc. 43rd IEEE Midwest Symposium on circuits and system Learning. August 2000.